Diplomarbeit:

Anwendung von Wavelets für die biometrische Authentikation

Martin Johns

Niedernstrasse 121
20095 Hamburg
mj@martinjohns.com

02. Februar 2003

Betreuer:

Prof. Dr. K. Brunnstein
und
Prof. Dr.-Ing. H. Siegfried Stiehl
Universität Hamburg
Fachbereich Informatik

mit Unterstützung von
Dipl. Inform. A. Brömme

Erklärung:

Ich versichere, dass ich die vorstehende Arbeit selbstständig und ohne fremde Hilfe angefertigt und mich anderer als der im beigefügtem Verzeichnis angegebenen Hilfsmittel nicht bedient habe. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen entnommen wurden, sind als solche kenntlich gemacht.

Hamburg, 2.2.2003 **Martin Johns**

Danksagung:

Mein aufrichtiger Dank gilt den Betreuern dieser Arbeit sowie allen, die mir hierbei mit Rat und Tat zur Seite standen:

Prof. Dr. Klaus Brunnstein
Prof. Dr.-Ing. H. Siegfried Stiehl
Dipl. Inf. Arslan Brömme
Frauke Wilck
Nikla Duffy
Hilmar Schulz
Dagmar Brunow
Anne Johns
und den Mitgliedern der Biometrikgruppe der Universität Hamburg.

Inhaltsverzeichnis

1 Einleitung **13**

2 Grundlagen **15**

2.1 Mathematische Grundlagen 15

2.1.1 Funktionenräume 15

2.1.2 Die Fouriertransformation 17

2.1.3 Die gefensterte Fouriertransformation 23

2.1.4 Grundlagen der Wavelets 25

2.1.5 Gaborwavelets 29

2.1.6 Die kontinuierliche Wavelettransformation 32

2.1.7 Die diskrete Wavelettransformation 33

2.1.8 Zweidimensionale Ansätze 41

2.1.9 Anwendungen von Wavelets 45

2.2 Authentikation 46

2.2.1 Authentikation durch Wissen 47

2.2.2 Authentikation durch Besitz 47

2.2.3 Authentikation durch Ort und Zeit 48

2.2.4 Authentikation durch biometrische Merkmale 48

2.2.5 Authentikationsmodelle 48

2.3 Grundlagen der biometrischen Methoden 49

2.3.1 Grundlegende Begriffe 49

2.3.2 Biometrische Prozesse 51

2.3.3 Leistungskenngrößen 54

2.3.4 Ausgewählte biometrische Techniken 56

3 Anwendungen von Wavelets 59
3.1 Wavelets in der Iriserkennung 59
3.1.1 Der Algorithmus von John Daugman 59
3.1.2 Iriserkennung nach Zhu et al. 68
3.1.3 Weitere Anwendungen in der Iriserkennung 70
3.2 Wavelets in der Fingerabdruckerkennung 70
3.2.1 Fingerabdruckanalyse von Y. Hamamoto 70
3.2.2 Weitere Anwendungen in der Fingerabdruckerkennung 75
3.3 Wavelets in der Gesichtserkennung 76
3.3.1 Lokalisierung von charakteristischen Punkten 76
3.3.2 Gesichtserkennung mit „Wavelet Packets“ 81
3.3.3 Weitere Anwendungen in der Gesichtserkennung . . . 85

4 Implementierung und Parametrisierung 87
4.1 Anforderungen an die Implementierung des Algorithmus . . . 88
4.1.1 Eigenschaften biometrischer Algorithmen 88
4.1.2 Eigenschaften waveletbasierter Algorithmen 89
4.2 Unterstützung der Implementierung 90
4.2.1 Das Rahmenwerk 91
4.2.2 Ablauf der Entwicklung 92
4.2.3 Der Vorgang der biometrischen Authentikation 94
4.3 Modularisierung des zu implementierenden Algorithmus . . . 96
4.3.1 Vorverarbeitung (P-Modul) 96
4.3.2 Qualitätssicherung und Normalisierung (Q-Modul) . . 96
4.3.3 Signaturberechnung (S-Modul) 97
4.3.4 Vergleich (D-Modul) 97
4.4 Einschränkung der Freiheitsgrade 97
4.4.1 Identifizierung der Freiheitsgrade 98
4.4.2 Algorithmus-inhärente Einschränkungen 99
4.4.3 Weitere Einschränkung der Freiheitsgrade 101
4.5 Implementierung 109
4.5.1 Auswahl der Entwicklungsumgebung 109
4.5.2 Umfang der Implementierung 112
4.5.3 Implementierung der Module 112

4.5.4 Routinen zur Steuerung und Auswertung 115
4.6 Experimente 115
4.6.1 Vorbereitung der Experimente 115
4.6.2 Verlauf der Experimente 117

5 Auswertung 125
5.1 Auswertung der Versuche 125
5.1.1 Resultate der Vorüberlegungen 125
5.1.2 Interpretation der Versuchsergebnisse 127
5.2 Theoretische Betrachtungen 130
5.2.1 Vergleich von Waveletkoeffizienten 130
5.2.2 Syntaktische Analyse 130
5.2.3 Vergleichsverfahren 131
5.2.4 Versuch einer Klassifikation 131
5.2.5 Unterschied zu „klassischen“ Verfahren 132

6 Zusammenfassung & Ausblick 135
6.1 Zusammenfassung der Resultate 135
6.2 Ausblick 137
6.2.1 Fortführung der praktischen Arbeit 137
6.2.2 Weiterführende Aspekte 139

A Mathematischen Notation i
A.1 Anpassung der Schreibweise i
A.2 Liste verwendeter mathematischer Symbole ii

B Weitere mathematische Grundlagen iii
B.1 Lineare Algebra iii
B.1.1 Vektorräume iii
B.1.2 Norm und Metrik iv
B.2 Faltung v
B.3 Statistische Methoden vi

C Details der Implementierng **viii**
C.1 P-Modul . . . viii
C.1.1 loadIrisImage.m . . . viii
C.1.2 getGrayImage.m . . . ix
C.2 Q-Modul . . . ix
C.2.1 getIrisBoundarys.m . . . ix
C.2.2 irisTransform.m . . . x
C.3 S-Modul . . . xi
C.3.1 getSignature.m . . . xi
C.3.2 mygabortransform.cpp . . . xii
C.3.3 generateIriscode.m . . . xiii
C.4 D-modul . . . xiii
C.4.1 hamming_abstand_mit_bitmasken.m . . . xiii
C.4.2 gesamtauswertung.m . . . xiv
C.5 Routinen zur Ablaufsteuerung . . . xvi
C.5.1 rahmen.m . . . xvi

D Dokumentation der verwendeten Irisbilder **xix**

E Details der Testergebnisse **xxi**
E.1 Grobe Abschätzung . . . xxiii
E.1.1 Einzelergebnisse . . . xxiii
E.2 Feine Abschätzung . . . xxvii
E.2.1 Gesamtergebnis . . . xxvii
E.2.2 Einzelergebnis 1 . . . xxviii
E.2.3 Einzelergebnis 2 . . . xxxiii
E.2.4 Einzelergebnis 3 . . . xxxviii
E.3 Untersuchung des Verhältnisses ω zu α . . . xliii
E.3.1 Gesamtergebnis . . . xliii
E.3.2 Einzelergebnis 1 . . . xliv
E.3.3 Einzelergebnis 2 . . . xlix
E.4 Untersuchung des Verhältnisse α zu β . . . liv
E.4.1 Gesamtergebnis . . . liv
E.4.2 Einzelergebnis 1 . . . lv
E.4.3 Einzelergebnis 2 . . . lx
E.4.4 Einzelergebnis 3 . . . lxv

Abbildungsverzeichnis

2.1 Einschränkung eines Signals 23
2.2 Die Gaußfunktion (ein- und zweidimensional) 25
2.3 Drei verschieden skalierte und verschobene Elemente einer Waveletfamilie 27
2.4 Das „Mexican Hat“-Wavelet mit zugehöriger Fouriertransformation 29
2.5 Das Haarwavelet und die zugehörige „Scalingfunction“ 39
2.6 Zweidimensionale Haar-„Scalingfinction“ und Haar-Wavelets $(\phi, \psi_1, \psi_2, \psi_3)$ 43
2.7 Zweidimensionales Haarwavelet nach [Bäni01] 44
2.8 Zweidimensionales Gaborwavelet (Real- und Imaginärteil) . . 45
2.9 Bestimmung der EER [Froe03] 56

3.1 Transformation des Irisrings 63
3.2 Ein transformierter Irisring 64
3.3 Visualisierung der Analyse-Bänder (aus [Daug99]) 66
3.4 Abbildung der Waveletkoeffizienten auf den Iriscode [Daug93] 66
3.5 Ein Fingerabdruck mit eingezeichnetem „Core Point“ und „Sampling Points“ [Hama99] 71
3.6 Schematische Darstellung eines Jets, bestehend aus zwölf Gaborwavelets [Wisk99]. 77
3.7 „Face Bunch Graph“, ohne und mit Markierung des „Local Expert“ [Wisk99]. 79
3.8 Zweistufige Analyse mit „Wavelet Packets“ [Garc00] 82
3.9 Identifizierung der Regionen des Gesichts über Koeffizientenmaxima [Garc00] 83
3.10 Die Regionen des Gesichts [Garc00] 84

3.11 Darstellung der Ergebnisse des zweiten Transformationsschritts [Garc00] 85

4.1 Iterative Verfeinerung des Algorithmus‘ [Broe02b] 92
4.2 Der Vorgang der biometrischen Authentikation (mit Datensammlung) [Broe02b] 95
4.3 Testbild zur Untersuchung der DC-Sensitivität 102
4.4 DC-Sensitivität bei einem Verhältnis von $\omega \cdot \alpha = 4$ 103
4.5 DC-Sensitivität bei einem Verhältnis $\omega \cdot \alpha = 6$ 103
4.6 Gaborwaveletfamilie mit einem konstanten Produkt von $\omega \cdot \alpha = 6$ (eindimensionale Darstellung) 104
4.7 Gaborwaveletfamilie mit einem konstanten Produkt von $\omega \cdot \alpha = 4$ (eindimensionale Darstellung) 105
4.8 DFT von Irisdaten 106
4.9 DFT von transformierten Irisdaten 106
4.10 Visualisierng der Bestimmung der Grenze zwischen Iris/Sklera über zwei Radien 113
4.11 Ergebnis der groben Abschätzung des Wertebereichs für ω . . 119
4.12 Ergebnis der feinen Abschätzung des Wertebereichs für ω . . 120
4.13 Ergebnis der Untersuchung des Verhältnisses ω zu α 121
4.14 Ergebnis der Untersuchung des Verhältnisses α zu β 123

5.1 Verwischte Irismuster durch fehlerhafte Vorverarbeitung . . . 127
5.2 Ergebnis der Versuche von John Daugman [Daug93] 128

D.1 Dokumentation der verwendeten Irisbilder xx

E.1 Einzelergebnis 1 der feinen Abschätzung xxviii
E.2 Einzelergebnis 2 der feinen Abschätzung xxxiii
E.3 Einzelergebnis 3 der feinen Abschätzung xxxviii
E.4 Einzelergebnis 1 der Untersuchung α/ω xliv
E.5 Einzelergebnis 2 der Untersuchung α/ω xlix
E.6 Einzelergebnis 1 der Untersuchung α/β lv
E.7 Einzelergebnis 2 der Untersuchung α/β lx
E.8 Einzelergebnis 3 der Untersuchung α/β lxv

Kapitel 1

Einleitung

In den vergangenen Jahren hat das Gebiet der Biometrik stetig an Bedeutung zugenommen. Es verging kaum eine Woche des Jahres 2002, in der nicht eine neue Nachricht über angedachte, geplante oder durchgeführte Einsätze biometrischer Techniken durch die Medien ging. Ob auf Flughäfen, in Sportstadien oder in Ausweisen: biometrische Methoden sind auf dem Weg ein fester Bestandteil des menschlichen Alltags zu werden.

Je größer die Bedeutung einer Technik wird, je mehr Sicherheitsmechanismen auf ihr beruhen und je zahlreicher die Werte werden, die durch diese Technik geschützt werden, desto stärker wächst die Motivation, diese Technik zu überwinden. Eine genaue Evaluation der verwendeten Technik ist deswegen unabdingbar.

Eine erfolgreiche Evaluation einer Technik, besonders einer biometrischen Technik, setzt ein tiefgehendes Verständnis dieser Technik voraus. Dieses Verständnis sollte alle Aspekte der Methode umfassen, die der Technik zugrunde liegt. Im Falle der Biometrik bzw. der in dieser Arbeit behandelten, biometrischen Authentikation beginnt dieses bei den Grundlagen des verwendeten Algorithmus, geht über die Besonderheiten einer Anwendung und Implementierung und reicht bis zu der potentiellen Güte der Technik.

Die Methoden der Biometrik sind zahlreich und mannigfach. Diese Arbeit beschäftigt sich mit Methoden die „Wavelets“ verwenden. Die Theorie der Wavelets entstammt den Gebieten der Mathematik und der Signalverarbeitung, die unabhängig von einander die Möglichkeiten dieser Theorie entdeckt haben. Wavelets erlauben eine umfassende, flexible und detaillierte Untersuchung von Daten. Im Gebiet der biometrischen Authentikation wurden sie erstmals von John Daugman zur Iriserkennung eingesetzt [Daug93]. In den folgenden Jahren nahm die Zahl waveletbasierter Anwendungen in der Biometrik stetig zu. Die Bandbreite der Anwendungsgebiete umfasst dabei derzeitig annähernd das gesamte Spektrum der biometrischen Techniken.

Mit dieser Arbeit soll ein umfassendes Wissen über die Anwendungen von

Wavelets in der biometrischen Authentikation vermittelt werden. Kapitel 2 enthält eine Einführung in die mathematischen Grundlagen der Wavelets. Diese Einführung wurde so verfasst, dass es auch Lesern ohne Vorkenntnisse im Bereich der Frequenzanalyse möglich sein sollte, das Fundament der Wavelettheorie zu verstehen. Grundkenntnisse der höheren Mathematik (insbesondere der linearen Algebra) werden dennoch vorausgesetzt. Fortgeschrittene Aspekte der Wavelettheorie, die wenig oder keine Anwendung in der Biometrik erfahren, werden nicht in ihrer vollen Tiefe behandelt. Der interessierte Leser sei für weitere Studien auf die zitierte Literatur verwiesen.

Weiterhin enthält das Kapitel 2 eine Einführung in die Gebiete der Authentikation im Allgemeinen (siehe Abschnitt 2.2) und der biometrischen Authentikation im Besonderen (siehe Abschnitt 2.3).

Kapitel 3 stellt ausgewählte Anwendungen von Wavelets in der biometrischen Authentikation vor. Insgesamt fünf verschiedene Algorithmen, verteilt auf die Gebiete der Iris-, Fingerabdruck- und Gesichtserkennung, werden detailliert beschrieben, wobei den Teilaspekten der Algorithmen, die Waveletmethoden verwenden, eine besondere Aufmerksamkeit zukommt. Die Zielsetzung dieses Kapitels ist es, eine Übersicht über die verschiedenen Möglichkeiten des Einsatzes von Wavelets zu erhalten.

Wie oben schon angedeutet, ist für ein tiefes Verständnis eines biometrischen Verfahrens, neben den algorithmischen Grundlagen, auch Wissen notwendig, das der Implementierung und dem praktischen Einsatz entstammt. Mit diesem Aspekt beschäftigt sich Kapitel 4: Anhand eines ausgewählten Beispielalgorithmus (John Daugmans Methode der Iriserkennung) wird exemplarisch vorgeführt, wie eine Methode zur biometrischen Authentikation konzipiert, implementiert und parametrisiert wird. Dabei wird den Eigenschaften, die aus der Verwendung von Wavelets erwachsen, besondere Aufmerksamkeit geschenkt.

Die separat erzielten Ergebnisse der Kapitel 3 und 4 werden im folgenden Kapitel 5 ausgewertet und zueinander in Beziehung gesetzt. Dabei wird versucht, algorithmus-übergreifende Eigenschaften waveletbasierter Verfahren zur biometrischen Authentikation aufzuzeigen.

Mit einer Zusammenfassung und einem Ausblick bildet das Kapitel 6 den Abschluss der Arbeit.

Kapitel 2

Grundlagen

2.1 Mathematische Grundlagen

2.1.1 Funktionenräume

Im Feld der Wavelettheorie basieren viele der mathematischen Zusammenhänge auf den Begriffen *Funktion* und *Signal*. Aus dem Bereich der reinen Mathematik stammend, definiert sich der Begriff der Funktion wie folgt [Köni01]:

Definition 2.1 (Funktion) *Unter einer reellen Funktion auf $\mathbb{R}$ versteht man eine Vorschrift f, die jedem $x \in \mathbb{R}$ in eindeutigerweise einen Wert $f(x) \in \mathbb{R}$ zuordnet.*

Die Menge $\mathbb{R}$ heißt Definitionsbereich, die Menge $f(\mathbb{R}) := \{f(x) \in \mathbb{R} \mid x \in \mathbb{R}\}$ Wertebereich von f. Die Teilmenge des Definitionsbereichs $T_f := \{x \in \mathbb{R} \mid f(x) \neq 0\}$, auf der f nicht verschwindet, nennt sich Träger von f.

Analog ist eine komplexe Funktion eine Vorschrift mit $f(x) \in \mathbb{C}$ für alle x.

Definitionen des Begriffs Signal, die man in Büchern aus dem Bereich der Systemtheorie findet, sind meist von dem physikalische Ursprung der Begriffsbildung geprägt. Dieses gilt auch für die folgende, aus [Kien98] stammende, Definition:

Definition 2.2 (Signal (1)) *Unter dem Begriff des Signals $f(t)$ versteht man den sich zeitlich verändernden Verlauf einer physikalischen Größe, die eine Information enthält.*

Die Signal-Definition aus [Giro97] schlägt den Bogen zum Funktionsbegriff:

Definition 2.3 (Signal (2)) *Ein Signal ist eine Funktion oder eine Wertefolge, die Informationen repräsentiert.*

Bemerkung 2.1 (Funktionen und Signale)
Die in den folgenden Abschnitten behandelte Theorie der Fourier- und Wavelettransformation fußt sowohl auf der reinen Mathematik wie auch auf der Signalverarbeitung. Dieses spiegelt sich auch in der verwendeten Literatur wieder, die Werke beider Disziplinen umfasst. Die Notation einiger Sachverhalte, die in den Werken beider Disziplinen behandelt werden, ist je nach verwendeter Quelle unterschiedlich. Einer der häufigsten und auffälligsten Unterschiede ist die Verwendung der Begriffe Funktion bzw. Signal. Beide Begriffe wurden soeben mathematisch definiert. Im Rahmen dieser Arbeit sind sie austauschbar. An jeder Stelle des Textes, in der von Signalen die Rede ist, greift ebenso der Begriff der Funktion beschränkter Energie (siehe Definition 2.4). Die Verwendung des jeweiligen Begriffs ergibt sich aus dem Zusammenhang oder der verwendeten Quelle. Es wurde stets der Begriff gewählt, der den Sachverhalt, den es zu erklären gilt, besser trifft bzw. zu dessen Verständnis beiträgt.

In Abschnitten, die in Zusammenhang mit der Fourieranalyse stehen, wird in den meisten Fällen die Notation der Signaltheorie verwendet. Dies ermöglicht die Verwendung des Vokabulars der Frequenzanalyse, mit Begriffen wie beispielsweise der Gegenüberstellung von *Zeit- und Frequenzraum*. Eine Ersetzung dieser Notation mit Begriffen des Räumlichen (im englischen „spatial“) führt in der Deutschen Sprache zu ungeschickten Konstruktionen.

In Abschnitten, die sich mit der Verarbeitung zweidimensionaler Bilddaten beschäftigen, ist es naheliegender den Begriff der Funktion zu verwenden, da eine Abbildung einer oder beider Koordinaten eines Bildes auf eine Zeitabhängigkeit nicht unmittelbar intuitiv ist.

Der Raum der Funktionen beschränkter Energie

Im Laufe der Arbeit werden wir uns mit Funktionen und Signalen von beschränkter Energie beschäftigen. Hierbei kann es sich um Funktionen handeln, die lediglich auf einem bestimmten Teil ihres Definitionsbereich nicht verschwinden oder für $\pm\infty$ hinreichend schnell gegen Null gehen. Der Raum dieser Funktionen ist von Interesse, da sich für ihn die Eigenschaften, die in den Abschnitten 2.1.2 bis 2.1.9 beschrieben werden, nachweisen lassen.

Mathematisch definiert sich der Raum der Funktionen beschränkter Energie folgendermaßen:

Definition 2.4 (Der Raum $L^2(\mathbb{R})$) *Eine Funktion $f(t)$ ist Element von $L^2(\mathbb{R})$ genau dann, wenn gilt:*

$$\int_{-\infty}^{+\infty} |f(t)|^2 \, dt < \infty.$$

Der Raum $L^2(\mathbb{R})$ ist ein $\mathbb{R}$-Vektorraum. Alle endlichen Signale sind Elemente diese Raums.

Für spätere Betrachtungen benötigen wir weiterhin eine Norm (siehe Anhang B.3) auf diesem Raum:

Satz 2.1 (L^2-Norm) *Mit*

$$\|f\| = \sqrt{\int_{-\infty}^{+\infty} |f(t)|^2 \, dt}$$

ist eine Norm auf den Raum $L^2(\mathbb{R})$ definiert.

Wenn nicht anders angegeben, wird im weiteren Verlauf dieser Arbeit stets die L^2-Norm verwendet.

Ein weiterer Funktionenraum, der im Abschnitt über die Fouriertransformation (2.1.2) auftritt, ist der Raum der integrierbaren Funktionen $L^1(\mathbb{R})$:

Definition 2.5 (Der Raum $L^1(\mathbb{R})$) *Ein Funktion $f(t)$ ist Element von $L^1(\mathbb{R})$ genau dann, wenn gilt:*

$$\int_{-\infty}^{+\infty} |f(t)| \, dt < \infty.$$

2.1.2 Die Fouriertransformation

Ein seit langem in der Signalverarbeitung fest verankertes Analysewerkzeug ist die *Fouriertransformation.*

Die Fouriertransformation beruht auf der Erkenntnis, dass die Menge der harmonischen Funktionen eine Basis für den Raum der Funktionen mit endlichem Integral bilden, d. h. alle Elemente dieses Raums können mit Hilfe von harmonischen Funktionen exakt dargestellt werden.

Ihren Ursprung hatte die Theorie 1807, als Jean-Baptiste Fourier nachwies, dass beliebige periodische Funktionen aus einer Reihe von Sinus- und Cosinus-Funktionen gebildet werden konnten. In den folgenden Jahrzehnten konnte die Theorie auf aperiodische Funktionen erweitert werden.

D. h., für jede Funktion mit endlichem Integral existiert eine Darstellung, die auf harmonischen Funktionen basiert. In Räumen mit endlicher Basis (also Räume von endlicher Dimension) können Elemente als Summe von Basiselementen dargestellt werden. Der betrachtete Funktionenraum besitzt unendlich viele Dimensionen.

Die komplexwertige Fourierreihe

Sowohl Vorläufer als auch Sonderfall der Fouriertransformation ist die komplexwertige *Fourierreihe*. Mit ihr werden periodische Funktionen $f(t)$ mit einer Periodenlänge von T als gewichtete Summe harmonischer Funktionen dargestellt.

Für eine periodischen Funktion $f(t)$ mit der Periodenlänge $T \in \mathbb{R}$ gilt:

$$f(t) = \sum_{k=-\infty}^{+\infty} c_k \, e^{ik\omega_0 t} \tag{2.1}$$

mit den komplexen Koeffizienten:

$$c_k = \frac{1}{T} \int_0^T x(t) e^{-ik\omega_0 t} \, dt \tag{2.2}$$

und

$$\omega_0 = \frac{2\pi}{T}, \tag{2.3}$$

wobei

$$c_0 = \frac{1}{T} \int_0^T x(t) \, dt \tag{2.4}$$

gilt.

Die Fouriertransformation

Für nicht periodische Funktionen kann die Fourierreihe nicht angewendet werden. Eine Verallgemeinerung ist aber möglich, wenn man die Grenzen der Periode T gegen $\pm\infty$ streben lässt. Die Bedingung für die Zulässigkeit dieses Grenzübergangs ist die absolute Integrierbarkeit der Funktion $f(t)$. Es muss daher gelten:

$$\int_{-\infty}^{+\infty} |t| \, dt < \infty, \tag{2.5}$$

d. h. $f(t)$ muss also Element des Raums $L^1(\mathbb{R})$ sein.

Bei dem Grenzübergang zu aperiodischen Funktionen ändert sich die Reihendastellung in eine Integraldarstellung:

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{f}(\omega) e^{i\omega t} \, d\omega \tag{2.6}$$

mit

$$\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(t) e^{-i\omega t} \, dt. \tag{2.7}$$

$\hat{f}(\omega)$ heißt die *Fouriertransformierte* von $f(t)$.

Frequenzen

Wie oben beschrieben, ist die Fouriertransformation ein Werkzeug, die ein Element des Raums $L^1(\mathbb{R})$ in eine Darstellung bezüglich der aus der Menge der harmonischen Funktionen bestehenden Basis transformiert. Eine harmonische Funktion $h(t)$ ist über ihre *Phase* ξ, *Frequenz* ω und *Amplitude* A eindeutig bestimmt:

$$h(t) = Ae^{i\xi} e^{i\omega t}. \tag{2.8}$$

Die Frequenz ω misst den Grad der Oszillation der harmonischen Funktion, also die Menge der Schwingungen pro Zeitraum. Je höher die Frequenz ist, desto höher ist die Anzahl der Schwingungen.

In der Darstellung eines Signals zur Basis der harmonischen Funktionen tritt jede Frequenz ω genau einmal auf. Für eine Frequenz ω gibt die Amplitude A an, welches Gewicht die Frequenz in dem Signal hat, und die Phase ξ die Phasenverschiebung der Schwingung.

Die Fouriertransformierte $\hat{f}(\omega)$ ist eine komplexwertige Funktion. Für eine Frequenz ω liefert $\hat{f}(\omega)$ den zugehörigen komplexen *Fourierkoeffizienten*, der die Phase und die Amplitude der Frequenz ω in dem Signal angibt:

$$\hat{f}(\omega) = Ae^{i\xi}. \tag{2.9}$$

Aus diesem Grund spricht man bei der Fouriertransformation von einer Abbildung aus dem *Zeitbereich* in den *Frequenzbereich*[1], d.h ein Signal $f(t)$ wird auf die Darstellung $\hat{f}(\omega)$ abgebildet.

Bemerkung 2.2 (Die DFT)
Zur Untersuchung der Eigenschaften der verwendeten Bilddaten wird in Kapitel 4 das Werkzeug der diskreten zweidimensionale Fouriertransformation (DFT) verwendet. Dieses ist eine auf digitale Bilder angewandte Form der Fouriertransformation[2], die neben Informationen über den Frequenzgehalt des Bildes auch Informationen über die Richtung dieser Frequenzen angibt. Die mathematischen Hintergründe der DFT sollen hier nicht weiter erörtert werden. In [Bovi00, S. 57ff] findet sich eine ausführliche Beschreibung dieser Methode.

Lokalisierung im Zeit-/Frequenzbereich

Der Begriff der Lokalisierung einer Funktion im Zeit- bzw. im Frequenzbereich bezieht sich auf den Bereich einer Funktion, in dem sie in dem jeweilig betrachteten Raum, also Zeit- oder Frequenzraum, nicht verschwindet. Je kleiner das Intervall, auf dem die Funktion nicht verschwindet, ist, desto besser ist die Funktion in diesem Raum lokalisiert[3].

Das Verhältnis der Lokalisierung einer Funktion im Zeitbereich zu ihrer Lokalisierung im Frequenzbereich ist umgekehrt proportional. Je besser eine Funktion also im Zeitraum zu lokalisieren ist, desto schlechter ist ihre Lokalisierung im Frequenzraum (und vice versa).

Beispiel 2.1 (Die Sinuswelle)
Die Sinuswelle erstreckt sich über den gesamten Zeitbereich. Ihre Fourier-Reihendarstellung hat andererseits lediglich einen Koeffizienten, der ungleich Null ist. Somit ist sie ausgezeichnet im Frequenzraum lokalisiert.

Beispiel 2.2 (Der Dirac-Stoß)
Der Dirac-Stoß $\delta(t-u)$ ist hervorragend im Zeitbereich lokalisiert, da er bis auf den Punkt $t = u$ für alle Werte von t verschwindet. Der Träger

[1] In diesem Zusammenhang spricht man auch vom Zeit- bzw. Frequenz*raum*.

[2] Die DFT ist eine Form der Fourierreihe: Für die Analyse wird angenommen, dass sich das Bild in jede Richtung unendlich oft wiederholt. Durch diese Konstruktion wird das Bild zu einer zweidimensionalen periodischen Funktion erweitert, auf welche die Methoden der Fourierreihe angewendet werden können.

[3] Genau genommen ist die oben formulierte Bedingung für eine gute Lokalisierung zu stark gefasst. Es reicht aus, den Bereich zu betrachten, in dem die Funktion nahezu verschwindet. Wenn dieser groß ist, sprechen wir von einer guten Lokalisierung. Z. B. verschwindet die Gaußfunktion e^{t^2} im Zeitbereich für kein $t \in \mathbb{R}$; dennoch ist sie verhältnismäßig gut im Zeitbereich lokalisiert.

der Fouriertransformierten des Dirac-Stoßes (siehe Anhang A.2) seinerseits erstreckt sich über die gesamte Breite des Frequenzspektrums.

Das umgekehrt proportionale Verhältnis von Zeit- zu Frequenzlokalisierung lässt sich wie folgt näher betrachten: Um die effektive Breite einer Funktion $f(t)$ zu verkleinern, kann diese um einen Faktor $s < 1$ gestaucht werden:

$$f_s(t) = \frac{1}{\sqrt{s}} f(\frac{t}{s}), \tag{2.10}$$

wobei der Faktor $\frac{1}{\sqrt{s}}$ dafür sorgt, dass die Norm der Funktion von der Stauchung nicht beeinflusst wird. Betrachten wir nun die Fouriertransformierte von $f_s(t)$, so sehen wir, dass diese um den Faktor $1/s$ gestreckt ist:

$$\hat{f}_s(\omega) = \sqrt{s}\hat{f}(s\omega). \tag{2.11}$$

Eine verbesserte Lokalisierung im Zeitbereich hat also zwangsweise eine schlechtere Lokalisierung im Frequenzbereich zur Folge. [Mall98] spricht in diesem Zusammenhang von einer Anwendung der Heisenbergschen Unschärferelation.

Für eine Untersuchung des zeitabhängigen Frequenzgehalts eines Signals sind Funktionen von Interesse, die sowohl im Orts- als auch im Zeitbereich gut lokalisiert sind. Das oben beschriebene umgekehrt proportionale Verhältnis der beiden Eigenschaften zeigt, dass für gegebene Funktionen eine Verbesserung in beiden Bereichen zugleich nicht möglich ist. Folglich ist die Frage zu klären, wie eine Funktion beschaffen sein muss, um in beiden Bereichen möglichst gut lokalisiert zu sein.

Ein Maß, das die Lokalisierung einer Funktion im Zeitbereich beschreibt, ist ihre effektive Breite Δt („effectiv width“) [Daug85]. Diese ist definiert durch die Varianz ihrer Energie-Verteilung [Mall98, S. 33]:

$$(\Delta t)^2 = \frac{1}{\|f\|^2} \int_{-\infty}^{\infty} |f(t)|^2 (t - t_0)^2 \, dt, \tag{2.12}$$

wobei t_0 der Schwerpunkt der Funktion $f(t)$ ist.

Ein entsprechendes Maß $\Delta\omega$ für die effektive Bandbreite im Frequenzraum berechnet sich analog zu 2.12 unter Verwendung der Fouriertransformierten $\hat{f}(\omega)$:

$$(\Delta\omega)^2 = \frac{1}{2\pi\|f\|^2} \int_{-\infty}^{\infty} |\hat{f}(\omega)|^2 (\omega - \omega_0)^2 \, d\omega, \tag{2.13}$$

wobei ω_0 die mittlere Frequenz von f ist:

$$\omega_0 = \frac{1}{2\pi\|f\|^2} \int_{-\infty}^{\infty} |\hat{f}(\omega)|^2 \omega \, d\omega. \tag{2.14}$$

Als Kenngröße für eine insgesamt gute Lokalisierbarkeit bietet sich das Produkt $\Delta t \Delta\omega$ an. Je kleiner das Produkt, desto besser ist die Funktion in beiden Bereichen lokalisiert.

[Mall98] zeigt, dass für $\Delta t \Delta\omega$ eine untere Grenze existiert. Weiterhin nennt er die notwendigen Eigenschaften einer Funktion, die diese untere Grenze erreicht [Mall98, S. 33]:

Satz 2.2 (Unschärferelation) *Für alle Funktionen $f(t)$ gilt*[4]*:*

$$(\Delta t)^2 (\Delta\omega)^2 \geq \frac{1}{4}$$

Die Ungleichheit wird zur Gleichheit genau dann, wenn es $(t_0, \omega_0, a, b) \in \mathbb{R}^2 \times \mathbb{C}^2$ gibt, so dass

$$f(t) = a e^{i\omega_0 t} e^{-b(t-t_0)^2}.$$

Ein Beweis dieses Satzes findet sich in [Mall98].

Im zweidimensionalen Fall gelten für eine Funktion $f(x, y)$, die an dem Punkt (t_0, y_0) zentriert ist und deren Fouriertransformierte $\hat{f}(\omega, \xi)$ bei (ω_0, ξ_0) zentriert ist, folgende Unschärferelationen (nach [Daug85]):

$$(\Delta t)^2 (\Delta\omega)^2 = \frac{\left(\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} (t - t_0)^2 |f(t,y)|^2 dt dy \right) \left(\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} (\omega - \omega_0)^2 |\hat{f}(\omega, \xi)|^2 d\omega d\xi \right)}{2\pi \left(\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} |f(t,y)|^2 \, dt dy \right)^2} \geq \frac{1}{4} \tag{2.15}$$

bzw.

$$(\Delta y)^2 (\Delta\xi)^2 = \frac{\left(\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} (y - y_0)^2 |f(t,y)|^2 dt dy \right) \left(\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} (\xi - \xi_0)^2 |\hat{f}(\omega, \xi)|^2 d\omega d\xi \right)}{2\pi \left(\int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} |f(t,y)|^2 \, dt dy \right)^2} \geq \frac{1}{4}. \tag{2.16}$$

[4] In [Daug85] gibt John Dauman als untere Grenze einen Wert von $\frac{1}{4\pi}$ an. Dieser, von Mallat abweichende, Wert erklärt sich durch die zu (2.12) verschiedene, von Daugman verwendete Berechnung der effektiven Breite einer Funktion.

2.1.3 Die gefensterte Fouriertransformation

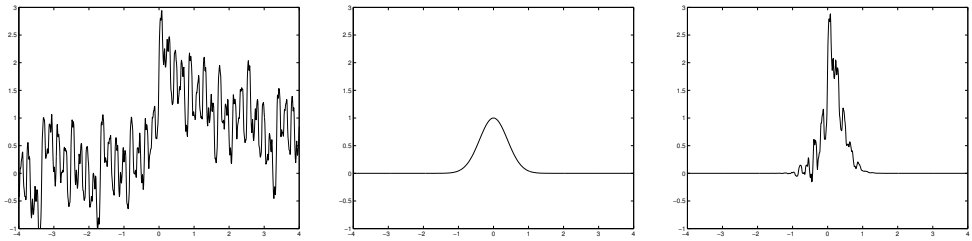
Die gefensterte Fouriertransformation bildet das Bindeglied zwischen der global wirkenden, d. h. nicht lokalisierten Fouriertransformation und der lokalisierten Wavelettransformation. Die Untersuchung von zeitlichen Signalen $f(t)$ mittels der Fouriertransformation hat den Vorteil, dass man Informationen über die im kompletten Signal vorkommenden Frequenzen in Form eines Spektrums erlangt, aber den Nachteil, dass kein lokales Spektrum bzgl. eines beliebigen Zeitpunkts t gewonnen werden kann.

Die gefensterte Fouriertransformation (auch „short time“ Fouriertransformation genannt) wurde erstmals 1946 von D. Gabor eingeführt [Mall98, S. 67], um zeitlich lokalisierte Frequenzanteile im Klang zu analysieren.

Die Idee, die dieser Methode der Analyse zugrunde liegt, ist die Einschränkung des Signals auf einen kurzen Zeitbereich mit anschließender Untersuchung des Frequenzgehalts in diesem eingeschränkten Bereich. Die Einschränkung des Signals erfolgt über eine so genannte *Fensterfunktion*. Eine Fensterfunktion $g(t)$ ist in ihrer Urform symmetrisch am Nullpunkt und verschwindet (bzw. verschwindet nahezu) bis auf einen definierten Bereich um den Nullpunkt herum, auf ihrem gesamten Definitionsbereich. Ihre Norm ist $\|g(t)\| = 1$.

Um ein Signal $f(t)$ auf einen Bereich um den Punkt u herum einzuschränken, wird das Zentrum der Fensterfunktion $g(t)$ auf u verschoben und das Signal mit der verschobenen Funktion multipliziert:

$$f_u(t) = f(t)g(t-u). \tag{2.17}$$



Signal $f(t)$ Fenster $g(t-u)$ Produkt $f_u(t)$

Abbildung 2.1: Einschränkung eines Signals

Die eingeschränkte Funktion $f_u(t)$ wird im Anschluss einer Fouriertransformation unterzogen.

$$\hat{f}_g(\omega, u) = \int_{-\infty}^{+\infty} f(t)g(t-u)e^{-i\omega t}\, dt. \tag{2.18}$$

Auf diese Weise können Informationen über die lokalen, d. h. die in der Umgebung um den Punkt u vorherrschenden, Frequenzen gewonnen werden. Die Art und Güte dieser Informationen hängt stark von der gewählten Fensterfunktion ab.

Wie die Fouriertransformation ist auch die gefensterte Fouriertransformation invertierbar, d. h. ein transformiertes Signal kann also verlustfrei aus der Transformierten zurückgewonnen werden. Neben der Frequenz kommt in diesem Fall als zweiter Parameter die Position des Fensters u hinzu [Bäni01, S. 163]:

$$f(t) = \frac{1}{2\pi\|g\|^2} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \hat{f}_g(\omega, u) g(t-u) e^{i\omega t} \, du \, d\omega. \tag{2.19}$$

Variable Fenstergröße

Eine Variation der Fenstergröße bietet die Möglichkeit, Signale unter Berücksichtigung ihres Zeitverlauf zu analysieren. So können Signalabschnitte mit gering schwankendem bzw. statischem Verlauf mit breiten Fenstern und Signalabschnitte, die sich durch eine starke Fluktuation der Funktionswerte auf engem Raum auszeichnen, mit schmalen Fenstern untersucht werden. Durch dieses Vorgehen können die definierenden Frequenzen für jeden Abschnitt des Signals ermittelt werden.

Die Fensterfunktion wird durch den Parameter s ergänzt, der die Größe des Fensters im Zeitbereich bestimmt:

$$g_{s,u} = \frac{1}{\sqrt{s}} g\left(\frac{t-u}{s}\right) \qquad \text{mit } s \in \mathbb{R}\backslash\{0\}. \tag{2.20}$$

Wahl der Fensters

Das Ergebnis einer gefensterten Fouriertransformation ist von der Wahl der verwendeten Fensterfunktion abhängig. Der Grad der zu erreichenden Genauigkeit in der Zeit-/Frequenz-Auflösung ist von dem Grad der Lokalisierung der Fensterfunktion in den respektiven Räumen (siehe 2.1.2) abhängig. Außerdem besitzen die verschiedenen Fensterfunktionen unterschiedliche, für sie charakteristische Frequenzeigenschaften (siehe [Mall98, S. 72ff] für eine detaillierte Vorstellung verschiedener Fensterfunktionen), die einen Einfluss auf das Ergebnis der Analyse haben. Unabhängig von der Wahl des Fensters gilt, je lokaler die Analyse im Zeitbereich ist (d. h. je kleiner das Fenster), desto weniger werden niederfrequente Signalanteile erfasst.

Eine häufig genutzte Fensterfunktion ist die Gaußfunktion:

$$g(t) = e^{-\frac{t^2}{2\sigma^2}}. \tag{2.21}$$

Die Gaußfunktion besitzt eine gute Lokalisierung in beiden Räumen. Durch den raschen Abfall ihrer Funktionswerte an den Flanken der Glockenkurve, gehen ihre Funktionswerte außerhalb eines bestimmten Intervalls sehr schnell gegen Null, was eine wertvolle Eigenschaft für numerische Berechnungen ist. Der Parameter ϕ bestimmt die Größe des Fensters im Zeitbereich. Im Falle der Analyse zweidimensionaler Daten (wie z. B. Bilddaten), wird eine zweidimensionale Gaußfunktion verwendet:

$$g(x, y) = e^{-(\frac{x^2}{2\phi^2}+\frac{y^2}{2\theta^2})}. \tag{2.22}$$

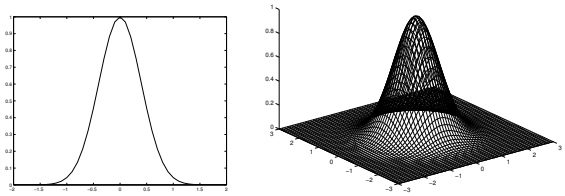


Abbildung 2.2: Die Gaußfunktion (ein- und zweidimensional)

2.1.4 Grundlagen der Wavelets

Die Wavelettransformation weist große Ähnlichkeit zu der gefensterten Fouriertransformation mit variabler Fenstergröße auf. Wie beschrieben, sind die harmonischen Funktionen, welche die Grundlage für die (gefensterte) Fouriertransformation bilden, schlecht im Zeitbereich lokalisiert. Aus diesem Grund wird die Analyse von Frequenzen bei kleiner werdendem Fenster (also genauerer Zeitbestimmung) ungenauer. Als Konsequenz wäre eine Analyse von Signalen durch Funktionen wünschenswert, die sowohl im Frequenz- wie auch im Zeitbereich gut lokalisiert sind.

Nach [Bäni01, S. 166] besteht die

> „[...] grundlegende Idee der Wavelet-Theorie [...] darin, dass der Faktor $e^{-i\omega t}$ und damit der Frequenzparameter ω in der Transformationsformel [der gefensterten Fouriertransformation] überflüssig werden, wenn [das Fenster] g selber schon eine gewisse Lokalisierung im Frequenzbereich aufweist“.

Wenn also die Fensterfunktion g eine vorherrschende Frequenz ω_0 aufweist, dann wird diese, wie in Abschnitt 2.1.2 beschrieben, durch den Skalierungsparameter s des Fensters beeinflusst. Wird das Fenster mittels Vergrößerung von s verbreitert, verkleinert sich die vorherrschende Frequenz; aus ω_0 wird $s^{-1}\omega_0$. Über eine Anpassung von s kann also jede Frequenz erreicht werden. Eine Funktion, wie sie Bäni beschrieben hat, würde also eine Analyse der Frequenzen in einem Signal ermöglichen, ohne dabei auf harmonische Funktionen zurück greifen zu müssen.

Die Wavelets bilden eine Familie von Funktionen, die den oben formulierten Ansatz verfolgen. Der Name Wavelet (deutsch: Wellchen, kleine Welle) leitet sich von der charakteristischen Form dieser Funktionen ab: Ein Wavelet ist eine wellenförmige, beschränkte Funktion, die außerhalb eines bestimmten Intervalls verschwindet. Wie im nächsten Abschnitt gezeigt wird, kann aus den Wavelets eine Basis des Raums $L^2(\mathbb{R})$ gebildet werden.

Definition 2.6 (Wavelet) *Eine normalisierte*[5]*, am Nullpunkt zentrierte Funktion $\psi(t) \in L^2(\mathbb{R})$ mit einem Integral von Null*

$$\int_{-\infty}^{+\infty} \psi(t)\, dt = 0 \tag{2.23}$$

heißt Wavelet.

Bemerkung 2.3 (Norm)
Wie schon in Abschnitt 2.1.1 angegeben, wird im weiteren Verlauf der Arbeit als Norm, wenn nicht anders erwähnt, die L^2-Norm verwendet.

Definition 2.7 (Waveletfamilie) *Aus einem so genannten Mutterwavelet $\psi(t)$ wird eine ganze Familie von Wavelets $\psi_{s,u}$ abgeleitet:*

$$\psi_{s,u}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right). \tag{2.24}$$

Diese Familie besteht aus Varianten des Mutterwavelets, die um den Faktor s skaliert und um den Faktor u zum Nullpunkt verschoben sind.

Durch den Faktor $\frac{1}{\sqrt{s}}$ bleibt die Skalierung normerhaltend. D. h., dass unabhängig von ihrer Skalierung alle Mitglieder einer Waveletfamilie dieselbe Norm besitzen.

Mit den Waveletfamilien wird eine Transformation definiert, die Ähnlichkeiten zu der Fouriertransformation besitzt. Bei dieser Transformation wird das Signal mit den skalierten und verschobenen Wavelets untersucht.

[5] Für eine Funktion bedeutet normalisiert $\|f(t)\| = 1$.

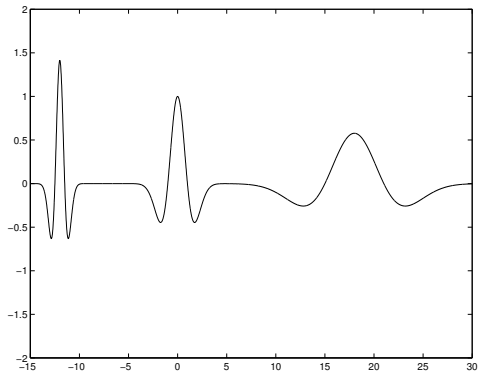


Abbildung 2.3: Drei verschieden skalierte und verschobene Elemente einer Waveletfamilie

Definition 2.8 (Wavelettransformation) *Sei $f(t) \in L^2(\mathbb{R})$. Die Berechnung*

$$Wf(u,s) = \int_{-\infty}^{+\infty} f(t)\psi^*_{u,s}(t)\,dt \tag{2.25}$$

wird Wavelettransformation genannt. $Wf(u,s)$ ist die Wavelettransformierte von $f(t)$.

Der Verschiebungsparameter u bestimmt den Punkt auf der Zeitachse, an dem der Analyseschritt vorgenommen wird. Der Skalierungsparameter s bestimmt den Bereich der Zeitachse, der von dem Analyseschritt betroffen ist. Das Ergebnis eines Schritts (also das Resultat der Berechnung von $Wf(u,s)$ für eine Belegung von u und s) nennt man *Waveletkoeffizient*.

Der folgende Satz gibt eine für die Analyse von Signalen wichtige Anforderung an die verwendeten Wavelets an [Mall98, S. 78]:

Satz 2.3 (Waveletzulässigkeitsbedingung) *Sei $\psi \in L^2(\mathbb{R})$ eine reelle Funktion, für die gilt:*

$$C_\psi = \int_0^{+\infty} \frac{|\hat{\psi}(\omega)|^2}{\omega} d\omega < +\infty, \tag{2.26}$$

dann bildet die Funktionenfamilie

$$\psi_{s,u}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right) \tag{2.27}$$

ein Erzeugendensystem für den Raum $L^2(\mathbb{R})$. Genauer gilt für alle $f(t) \in L^2(\mathbb{R})$:

$$f(t) = \frac{1}{C_\psi} \int_0^{+\infty} \int_{-\infty}^{+\infty} Wf(u,s)\psi_{s,u}(t)du\frac{ds}{s^2}. \tag{2.28}$$

Ein Beweis dieses Satzes findet sich in [Mall98]. Der Satz sagt aus, dass wenn eine Funktion die Waveletzulässigkeitsbedingung erfüllt, dann bildet die aus ihr gebildete Funktionenfamilie ein Erzeugendensystem (siehe Anhang B.2) für den $L^2(\mathbb{R})$. Ein Signal $f(t)$ kann vollständig aus der Transformierten zurückgewonnen werden. Dieses bedeutet, dass die Transformation von $f(t)$ in $Wf(u,s)$ verlustfrei erfolgt, also keine Informationen über das Signal bei der Transformation verloren gehen.

Folgerungen aus Satz 2.3: Aus der Waveletzulässigkeitsbedingung (Formel 2.26) lassen sich einige charakteristische Eigenschaften von Wavelets ableiten:

- $\hat{\psi}(0) = 0$, denn sonst entsteht eine Polstelle an dem Punkt $\omega = 0$, was eine Verletzung der Bedingung $C_\psi < +\infty$ zur Folge hätte. In (2.7) eingesetzt, ergibt sich

  $$0 = \hat{\psi}(0) = \int_{-\infty}^{+\infty} \psi(t)e^0\, dt = \int_{-\infty}^{+\infty} \psi(t)\, dt.$$

  Also ist das Integral aller Funktionen, welche die Waveletzulässigkeitsbedingung erfüllen, gleich null (eine Eigenschaft, die wir oben schon als für Wavelets definierend genannt haben).

- Bei Wavelets mit symmetrischer Fouriertransformierten ergibt sich eine bandpassartige Frequenzcharakteristik. Die Fouriertransformierte ist symmetrisch, hat eine Nullstelle am Nullpunkt und geht in den Richtungen $\pm\infty$ gegen Null. Somit gibt es ein klar abgegrenztes Intervall von Frequenzen, deren Anteil im Wavelet ungleich null sind. Alle Frequenzen außerhalb dieses Intervalls verschwinden.

Beispiel 2.3 („Mexican Hat“)
Ein Wavelet, das in der Bildverarbeitung häufig verwendet wird, ist das „Mexican Hat“-Wavelet. Es ergibt sich aus der zweiten Ableitung der Gauß-Funktion:

$$\psi(t) = \frac{2}{\pi^{\frac{1}{4}}\sqrt{3\sigma}} \left(1 - \frac{t^2}{\sigma^2}\right) e^{-\frac{t^2}{2\sigma^2}}. \tag{2.29}$$

Die Fouriertransformierte des „Mexican Hat“-Wavelet lautet:

$$\hat{\psi}(\omega) = \frac{-\sqrt{8}\sigma^{\frac{5}{2}}\pi^{\frac{1}{4}}}{\sqrt{3}}\omega^2 e^{-\frac{\sigma^2\omega^2}{2}}. \tag{2.30}$$

Abb. 2.4 zeigt für $\sigma = 1$ das Wavelet und seine Fouriertransformierte. An der Abbildung der Fouriertransformierten kann gut das bandpass-artige Frequenzspektrum des Wavelets abgelesen werden.

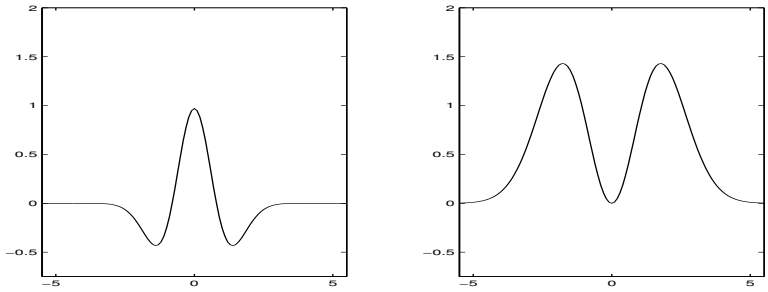


Abbildung 2.4: Das „Mexican Hat“-Wavelet mit zugehöriger Fouriertransformation

Bemerkung 2.4 (Variation und Stetigkeit)
Da ein Wavelet ψ ein Integral von Null besitzt, misst das Wavelet-Integral (siehe Formel 2.25) die Variation der Funktion $f(t)$ in der Nachbarschaft von u auf einem Intervall, dessen Größe proportional zu s ist [Mall98, S. 77]. Unter anderem kann daraus gefolgert werden, dass der Waveletkoeffizient, wenn s gegen Null geht, die Stetigkeit von $f(t)$ an der Stelle u misst. Diese Eigenschaft ist z. B. wichtig für Anwendungen zur Detektion von scharfen Übergängen im Signal (wie Kanten in Bildern[6]).

2.1.5 Gaborwavelets

Für die zeitabhängige Untersuchung der Entwicklung des Frequenzgehalts eines Signals benötigt man ein Wavelet, das die Phasen- von den Amplitudeninformationen eines Signale trennt. Für diese Aufgabe führt Mallat die analytischen Wavelets ein [Mall98, S. 81ff]:

Definition 2.9 (Analytische Funktion) *Eine Funktion $f_a \in L^2(\mathbb{R})$ heißt analytisch, wenn ihre Fouriertransformierte für negative Frequenzen verschwindet:*

[6] Die Arbeit [Schr00] vertieft speziell diesen Aspekt.

$$\omega < 0 \Rightarrow \hat{f}_a(\omega) = 0$$

Analytische Funktionen sind zwangsläufig komplex. Ein analytisches Wavelet ist ein Wavelet, auf das Definition 2.9 zutrifft. Die Wavelettransformation eines reellen Signals $f(t)$ mit analytischen Wavelets, welche die Bedingung 2.3 erfüllen, ist vollständig. Bei der Rekonstruktion eines reellen Signals aus den Waveletkoeffizienten wird der Imaginärteil der Rekonstruktion verworfen [Mall98]:

$$f(t) = \frac{1}{C_\psi} \text{Real} \left[\int_0^{+\infty} \int_{-\infty}^{+\infty} Wf(u,s)\psi_{s,u}(t) du \frac{ds}{s^2} \right]. \tag{2.31}$$

Hervorgegangen aus der gefensterten Fouriertransformation, konstruiert sich ein *Gaborwavelet* aus einem Gauß-Fenster, dass mit einer komplexen Welle der Frequenz ξ multipliziert wurde:

$$\psi(t) = e^{i\xi t}\, g(t) \tag{2.32}$$

mit

$$g(t) = \frac{1}{(\sigma^2\pi)^{\frac{1}{4}}} e^{-\frac{t^2}{2\sigma^2}}. \tag{2.33}$$

Die Fouriertransformierte von $\psi(t)$ laute $\hat{\psi}(\omega) = \hat{g}(\omega - \xi)$, mit einer Transformierten des verwendeten Gauß-Fensters von:

$$\hat{g}(\omega) = (4\pi\sigma^2)^{\frac{1}{4}} e^{-\frac{\sigma^2\omega^2}{2}}. \tag{2.34}$$

Somit gilt, falls das Produkt $\sigma^2\xi^2 \gg 1$ ist, dass für negative Frequenzen $\hat{\psi}(\omega) = \hat{g}(\omega - \xi) \approx 0$ gilt. Aus diesem Grund sind Gaborwavelets annähernd analytisch. Gaborwavelets erfüllen die im Satz 2.2 formulierte Bedingung und besitzen somit ideale Lokalisierung im Zeit- und Frequenzraum.

Gaborwavelets finden in verschiedenen Feldern der Signalverarbeitung breite Anwendung. Neben ihrer hervorragenden Lokalisierung im Zeit-/Frequenzraum werden für die Verwendung dieser Wavelets häufig biologische Motivationen angegeben: Untersuchungen des visuellen Kortexes des menschlichen Auges ergaben auffällige Parallelen zwischen dem Filterverhalten der Gaborwavelets und dem Verhalten spezifischer Zellen des Auges. Weitere Informationen zu diesem Thema enthalten u.a. die Artikel [Corm00], [Moul00], [Daug85] sowie [Siem91].

Bemerkung 2.5 (DC-Sensitivität)
Gaborwavelets in der oben angegebenen Form haben den Nachteil, dass sie *DC-sensitiv*[7] sind. Dieses heißt, dass die Filterantwort auf konstante, strukturlose Signale ungleich 0 ist. Der Betrag des reellen Anteils der Filterantwort ist proportional zu der lokalen Intensität der Bilddaten. Die Folgen dieser Eigenschaft beschreibt [Fröh00] mit der paradox erscheinenden Beobachtung, dass schwache, aber sichtbare Muster innerhalb eines hellen Bildes nicht identifiziert werden können, in einem dunklen Bild aber vergleichbare Muster zu einen Ergebnis führen, auch wenn diese für das menschliche Auge schwer oder gar nicht erkennbar sind.

Zur Vermeidung dieses inkonsistenten Verhaltens führt [Fröh00, S. 32] folgendes Glied ein, das die DC-Sensitivität ausgleichen soll:

$$e^{-\frac{1}{2}\omega_0^2\sigma_x^2}. \tag{2.35}$$

Daraus resultiert folgende Formel für die Gaborwavelets:

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma_x} e^{-\frac{x^2}{2\sigma_x^2}} \left(e^{i\omega_0 x} - e^{-\frac{1}{2}\omega_0^2\sigma_x^2} \right). \tag{2.36}$$

Bemerkung 2.6 (Notation)
Je nach Quelle existieren leichte Unterschiede in der genauen Notation der Gaborwavelets. So unterscheidet sich in manchen Fällen die Definition des Gauß-Fensters. So verwendet z. B. [Siem91]:

$$g(t) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2}\sigma t^2 \sigma^{-1}}. \tag{2.37}$$

Ebenso existieren Varianten in der genauen Spezifikation der mittleren Frequenz ξ: Je nach Quelle kann der Frequenzparameter einen positiven Vorfaktor besitzen, wie z. B. bei [Wisk99] (hier eine vereinfachte Darstellung des zweidimensionales Gaborwavelet, siehe auch Abschnitt 3.3.1):

$$\psi(x, y) = g(x, y)\, e^{i\xi_k x}. \tag{2.38}$$

Dieses führt zu einer Umkehrung der Polarität des Imaginärteils der Wavelets (da der Sinus eine ungerade Funktion ist). Eine weitere Variante verwendet [Pich97], in der ξ einem Vorfaktor von 2π besitzt:

$$\psi(t) = g(t)\, e^{i2\pi x}. \tag{2.39}$$

[7] Eine treffende Übersetzung dieses, dem englischsprachigen Werks [Fröh00] entnommenen, Begriffs konnte leider nicht gefunden werden. Aus diesem Grund wird der Begriff im weiteren Verlauf der Arbeit unverändert verwendet.

Für alle Varianten gilt, dass die in diesem Abschnitt beschriebenen mathematischen Eigenschaften der Gaborwavelets von den verschiedenen Ausprägungen unberührt bleiben.

Neben der hier vorgestellten komplexen Notation der Gaborwavelets existiert weiterhin eine Schreibweise, welche die Transformation nach Real- und Imaginärteil aufteilt. In diesem Fall werden aus dem komplexen Gaborwavelet zwei separate reelle Funktionen gewonnen. Der Artikel [Zhu00a] (siehe Abbschnitt 3.1.2) verwendet diese Methode. In diesem Fall wird häufig vom „Even“- und „Odd“-Anteil gesprochen.

Die Verwendung von Gaborwavelets hat sich unabhängig von der Wavelettheorie aus der gefensterten Fouriertransformation entwickelt. Aus diesem Grund sprechen einige Artikel, die Gaborwavelets verwenden, von „Gabor-Filtern“. In diesem Fall wird für die Berechnung eines Koeffizienten das Signal meist mit dem Gaborwavelet gefaltet (siehe Anhang B.2), im Gegensatz zu der Notation der Wavelettheorie, in der Signal und Wavelet miteinander multipliziert werden.

2.1.6 Die kontinuierliche Wavelettransformation

In Abschnitt 2.1.4 wurde eine Tranformation unter der Verwendung von Wavelets eingeführt (siehe Definiton 2.8). Im weiteren Verlauf dieses Abschnittes wurde erläutert, welchen Bedingungen die verwendeten Wavelets $\psi_{u,s}(t)$ genügen müssen (siehe Satz 2.3), damit eine Rekonstruktion des Signals aus der Wavelettransformierten möglich ist. Diese Transformation trägt den Namen *kontinuierliche Wavelettransformation*, da die beiden Werte u (Translation) und s (Skalierung) bei der Transformation kontinuierlich parametrisiert werden. Daraus resultierend ist $Wf(u,s)$ eine in beiden Parametern kontinuierliche Funktion:

$$Wf(u,s) = \int_{-\infty}^{\infty} f(t)\frac{1}{\sqrt{s}}\psi(\frac{t-u}{s})\,dt$$

Ein Signal $f(t)$ wird folgendermaßen aus seiner Wavelettransformierten zurückgewonnen:

$$f(t) = \frac{1}{C_\psi}\int_{0}^{+\infty}\int_{-\infty}^{+\infty} Wf(u,s)\psi_{s,u}(t)du\frac{ds}{s^2}$$

Die kontinuierliche Wavelettransformation hat folgende Eigenschaften:

- Translationsinvarianz

  Wenn ein Signal f um τ verschoben wird, man also zum Signal $f_\tau(t) = f(t-\tau)$ übergeht, verschiebt sich die Transformierte mit $Wf_\tau(s,u) = Wf(s,u-\tau)$ (siehe [Bäni01, S. 169]). Diese Eigenschaft garantiert, dass das Ergebnis einer Analyse nicht von der Wahl des Nullpunktes der Zeitskala abhängt.

- Redundanz

  Die kontinuierliche Wavelettransformation ist hoch redundant. Dieses zeigt sich in der Tatsache, dass eine eindimensionale Funktion $f(t)$ durch eine zweidimensionale Funktion $Wf(u,s)$ dargestellt wird. Der Grund für die Redundanz liegt darin, dass die Funktionenfamilie $\psi_{u,s}$ bei kontinuierlicher Parametrisierung lediglich ein Erzeugendensystem und keine Basis des $L^2(\mathbb{R})$ bildet. Der Informationsgehalt von $Wf(u,s)$ unterscheidet sich für kleine τ nur gering von $Wf(u-\tau,s)$ oder $Wf(u,s-\tau)$.

- Rechenaufwand

  Durch die Notwendigkeit einer kontinuierlichen Parametrisierung der Werte u und s über ihren gesamten Definitionsbereich, ist die kontinuierliche Wavelettransformation, im Vergleich zu beispielsweise der Fouriertransformation, aufwändig.

Anwendung findet diese Form der Transformation in Bereichen, in denen die Eigenschaft der Translationsinvarianz von Bedeutung ist. Dazu [Bäni01, S. 169]:

> „Die kontinuierliche Wavelet-Transformation und die damit erstellten Skalendiagramme sind vor allem dann nützlich, wenn es darum geht, lokale Merkmale zu erkennen; etwa ein Elektro-Encephalogramm (EEG) eines Epilepsie-Patienten nach bestimmten Wellenformen abzusuchen oder in einem Bild die Kanten zu bestimmen“.

2.1.7 Die diskrete Wavelettransformation

Wie im letzten Abschnitt gesehen werden konnte, hat die kontinuierliche Wavelettransformation den Nachteil sehr aufwändig zu sein, was mit der hohen Redundanz dieser Transformation zusammenhängt.

Eine naheliegende Möglichkeit besteht darin, die Redundanz und damit auch den Aufwand zu verringern, ist die Menge der zugrunde liegenden Funktionen zu verkleinern, die den betrachteten Raum aufspannen. Wie in Satz 2.3 beschrieben, bilden die Wavelets ein Erzeugendensystem (siehe Anhang

B.2) für den Raum $L^2(\mathbb{R})$. Mit einer Minimierung der Anzahl der Elemente dieses Erzeugendensystems geht eine Verringerung des Aufwands und der Redundanz einher.

Das Erzeugendensystem, das der kontinuierlichen Wavelettransformation zugrunde liegt, wird durch eine kontinuierliche Skalierung und Translation des Mutterwavelets erzeugt. Bei der *diskreten Wavelettransformation* entsteht das Erzeugendensystem durch eine diskrete Skalierung und Translation. Die Bedingung, dass die so entstandene Waveletfamilie weiterhin die gewünschten Eigenschaften hat, dass also eine Funktion $f(t)$ vollständig aus ihrer Wavelettransformierten rekonstruierbar ist, gibt die folgende Definition des *„Frames*[8]*“* an [Krüg01]:

Definition 2.10 („Frame“) *Sei $\psi \in L^2(\mathbb{R})$ ein Wavelet, $S \subset \mathbb{R}^2$ eine diskrete Menge von reellen Zwei-Tupeln und $B_\psi = \{\psi_{m,n} \mid (m,n) \in S\}$ eine diskrete Familie von Wavelets. B_ψ bilden einen „Frame“, wenn Konstanten $A > 0$ und $B < \infty$ existieren, so dass für alle $f(x) \in L^2(\mathbb{R})$ gilt:*

$$A\|f\|^2 \leq \sum_{(m,n)\in S} \left| \int_{-\infty}^{\infty} \psi_{(m,n)}(t) f(t)\, dt \right|^2 \leq B\|f\|^2. \tag{2.40}$$

A und B werden als „Frame“-Grenzen („frame bounds“) bezeichnet

Die Form der diskreten Wavelettransformation gleicht der Kontinuierlichen:

$$Wf(u,s) = \int_{-\infty}^{+\infty} f(t)\psi_{u,s}(t)\, dt, \tag{2.41}$$

mit der Einschränkung, dass $(u,s) \in S$ gilt.

Falls ein $S \subset \mathbb{R}^2$ existiert, so dass B_ψ einen „Frame“ für $L^2(\mathbb{R})$ bildet, dann kann jedes Element des $L^2(\mathbb{R})$ aus der zu B_ψ gehörigen Wavelettransformation exakt rekonstruiert werden.

Falls $A = B$ gilt, sagt man, dass der „Frame“ eng („tight“) ist. In diesem Fall kommen zur Rekonstruktion von $f(t)$ die Elemente von B_ψ zum Einsatz:

$$f(t) = \frac{2}{A+B} \sum_{(m,n)\in S} Wf(m,n)\psi_{m,n}(t). \tag{2.42}$$

Das Maß $\frac{A+B}{2}$ misst die Redundanz der Transformation. Für „Frame“-Grenzen $A = B = 1$ und $\|\psi\| = 1$ bildet B_ψ eine orthonormale Basis des

[8] In den meisten Werken werden Frames genereller definiert. Die angegebene Definition 2.10 ist auf den Spezialfall der durch Wavelets gebildeten Frames des $L^2(\mathbb{R})$ beschränkt.

$L^2(\mathbb{R})$ und die Darstellung in (2.42) ist eindeutig. Für „Frame“-Grenzen $A = B > 1$ ist die Gleichung (2.42) weiterhin exakt, aber nicht in jedem Falle eindeutig.

Im Falle von $A \neq B$ kann die Funktion nicht aus den Wavelets aus B_ψ rekonstruiert werden. Zur Rekonstruktion von $f(t)$ kommt eine zweite Waveletfamilie $\tilde{\psi}_{m,n}$ zum Einsatz:

$$f(t) = \sum_{(m,n)\in S} Wf(m,n)\tilde{\psi}_{m,n}. \tag{2.43}$$

Die Waveletfamilie $B_{\tilde{\psi}}$ heißt in diesem Fall *dualer „Frame“*.

Tiefer in die Materie der „Frame“-Theorie einzudringen, ist für das weitere Verständnis dieser Arbeit nicht nötig. Wichtig ist jedoch, dass die diskrete Wavelettransformation eine verlustfreie (und bei Verwendung geeigneter Waveletfamilien auch redundanzfreie) Transformation der Elemente des $L^2(\mathbb{R})$ bietet.

Orthogonale Waveletbasen

Im letzten Abschnitt wurden die notwendigen Anforderungen an Wavelets formuliert, die eine diskrete Waveletfamilie bilden. Wie schon erwähnt, resultiert aus einer diskreten Waveletfamilie nicht zwangsläufig eine redundanzfreie Transformation.

Für performante Anwendungen von Wavelets ist eine niedrige Redundanz essentiell. Aus diesem Grund sind Waveletfamilien, die eine orthogonale Basis bilden, von hohem Interesse.

Es wurde gezeigt, dass es möglich ist Wavelets so zu konstruieren, dass die Waveletfamilie der Form

$$\left\{\psi_{j,n}(t) = \frac{1}{\sqrt{2^j}}\psi\left(\frac{t-2^j n}{2^j}\right)\right\}_{(j,n)\in\mathbb{Z}^2} \tag{2.44}$$

eine orthogonale Basis des $L^2(\mathbb{R})$ bilden.

Wie der Formel (2.44) entnommen werden kann, findet sowohl die Skalierung als auch die Translation der Wavelets dieser Familie in Zweierpotenzschritten statt, wobei die Schrittgrößen der Skalierung und der Translation einander bedingen. Ein Wavelet der Skalierung 2^j wird in Schritten der Größe 2^j bezüglich des Signals verschoben. So wird ein Signal von vielen „kleinen“ und wenigen „großen“ Wavelets abgetastet. Die großen Wavelets untersuchen die groben Informationen des Signalverlaufs, die kleinen Wavelets die Details.

Diese Art der Verwendung der Wavelets schlägt den Bogen zu der im Bereich der Bild- und Signalverarbeitung verbreiteten Multiresolutions-Analyse, einer Methode, bei der die zu untersuchenden Daten unter verschiedenen, feiner werdenden Auflösungen betrachtet werden.

Die „Scalingfunction“

Entscheidend für die Konstruktion einer Waveletbasis, die der Gleichung (2.44) entspricht, ist die so genannte *„Scalingfunction“*. Die „Scalingfunction“ legt das mathematische Fundament einer orthogonalen Waveletfamilie.

Bevor wir die „Scalingfunction“ definieren können, muss zuerst die *Multiresolutions-Approximation*[9] definiert werden. Wie [Mall98, S. 225ff] erläutert, findet dafür eine Konstruktion von Unterräumen $V_j \subset L^2(\mathbb{R})$ statt, die folgenden Bedingungen genügt:

Definition 2.11 (Multiresolution) *Eine Sequenz $\{V_j\}_{j\in\mathbb{Z}}$ von abgeschlossenen Unterräumen des $L^2(\mathbb{R})$ ist eine Multiresolutions-Approximation, wenn gilt:*

1. $$\forall\,(j,k) \in \mathbb{Z}^2, f(t) \in V_j \;\Leftrightarrow\; f(t-2^j k) \in V_j \tag{2.45}$$
2. $$\forall\, j \in \mathbb{Z}, V_{(j-1)} \subset V_j \tag{2.46}$$
3. $$\forall\, j \in \mathbb{Z}, f(t) \in V_j \;\Leftrightarrow\; f\left(\tfrac{t}{2}\right) \in V_{j+1} \tag{2.47}$$
4. $$\lim_{j\to+\infty} V_j = \bigcap_{j=-\infty}^{\infty} V_j = \{0\} \tag{2.48}$$
5. $$\lim_{j\to-\infty} V_j = \text{Abschluss}\left(\bigcup_{j=-\infty}^{\infty}\right) = L^2(\mathbb{R}) \tag{2.49}$$
6. *Es existiert $\theta(t)$, so dass $\{\theta(t-n)\}_{n\in\mathbb{Z}}$ eine Riesz-Basis von V_0 ist.*

Der Name der Multiresolutions-Approximation entstammt dem Zusammenhang, dass eine Projektion eines Signals $f(t)$ in den Raum V_j eine Approximation der Genauigkeit j des Signals ergibt, wobei die Genauigkeit der Approximation mit abfallendem j zunimmt.

Ohne tiefer in die Theorie der Multiresolution einzudringen, schließen wir gleich folgenden Satz an, der die Orthonormalität der Basen der Räume V_j garantiert:

Satz 2.4 („Scalingfunction“) *Sei $\{V_j\}_{j\in\mathbb{Z}}$ eine Multiresolutions-Approximation und ϕ eine Funktion, deren Fouriertransformierte folgende Form hat:*

[9] [Schr00, S. 36] merkt an, dass [Mall98], der Quelle für diesen Abschnitt der Arbeit, auf eine Unterscheidung der Begriffe *Multiresolutions-Approximation* und *Multiskalen-Approximation* verzichtet. In dieser Arbeit wird die Notation von Mallat verwendet. Eventuelle Differenzen zu Begriffsdefinitionen anderer Werke sind somit möglich.

$$\hat{\phi}(\omega) = \frac{\hat{\theta}(\omega)}{\left(\sum_{k=-\infty}^{+\infty} |\hat{\theta}(\omega + 2k\pi)|^2\right)^{\frac{1}{2}}}. \tag{2.50}$$

dann bildet die Funktionenfamilie

$$\left\{\phi_{j,n}(t) = \frac{1}{\sqrt{2^j}}\phi\left(\frac{t-n}{2^j}\right)\right\}_{n\in\mathbb{Z}} \tag{2.51}$$

eine Orthonormalbasis des Raums V_j für alle $j \in \mathbb{Z}$. ϕ heißt „Scalingfunction“.

Ein Beweis dieses Satzes findet sich bei [Mall98, S. 225] Fassen wir die Aussagen der Definition 2.11 und des Satzes 2.4 zusammen: Mit den V_j wurde eine absteigende Reihe von Unterräumen des $L^2(\mathbb{R})$ definiert. Der Satz 2.4 garantiert uns, dass die Elemente der Basis des Raums V_j eine Orthonormalbasis bilden. Mit kleiner werdendem Index j wird die Auflösung des Raums V_j feiner. Jedes Element von $L^2(\mathbb{R})$ kann beliebig genau über Elemente der Räume V_j approximiert werden; beim Grenzübergang $j \to -\infty$ erhalten wir den Ausgangsraum $L^2(\mathbb{R})$.

Konstruktion der orthonormalen Wavelets

Betrachtet man die Funktionenfamilie aus Satz 2.4

$$\left\{\phi_{j,n}(t) = \frac{1}{\sqrt{2^j}}\phi\left(\frac{t-n}{2^j}\right)\right\}_{n\in\mathbb{Z}}$$

fällt auf, dass ihre Form eine starke Ähnlichkeit zu der angestrebten Waveletfamilie aus Gleichung 2.44 besitzt. Der Unterschied der $\phi_{j,n}$ zu den gesuchten Wavelets ist, dass die $\phi_{j,n}$ zwar für eine feste Skala j zueinander orthogonal sind, diese Eigenschaft aber nicht skalenübergreifend gilt ($\phi_{j,n} \in V_j$ und $\phi_{j,n} \in V_{j'}$ für alle $j' < j$).

Der Raum V_j ist in dem Raum V_{j-1} enthalten. Da aber $V_j \neq V_{j-1}$ gilt, existiert notwendigerweise ein Raum $W_j = V_{j-1} \setminus V_j$. W_j ist das orthogonale Komplement zu V_j:

$$V_{j-1} = V_j \oplus W_j. \tag{2.52}$$

Bildlich gesprochen enthält W_j die „Details“, die verloren gehen, wenn von $j-1$ zu der gröberen Skala j übergegangen wird.

Beispiel 2.4 (Konstruktion von Räumen) Lediglich durch Hinzufügen der Detailräume W_j, kann aus einem Raum V_j jeder feinere Raum $V_{j',j'<j}$ konstruiert werden:

$$\begin{aligned} V_4 &= V_5 \oplus W_5 \\ V_3 &= V_5 \oplus W_5 \oplus W_4 \\ V_2 &= V_5 \oplus W_5 \oplus W_4 \oplus W_3 \\ V_1 &= \cdots \end{aligned}$$

Für alle $j \in \mathbb{Z}$ gilt, $W_j \subset V_{j-1}$ und $W_{j-1} \bigcap V_{j-1} = \{0\}$. Daraus folgt $W_j \bigcap W_{j-1} = \{0\}$. Da W_j zu V_j orthogonal ist, ist W_j auch orthogonal zu W_{j+1}.

Der folgende Satz gibt den Zusammenhang zwischen der „Scalingfunction“ und der Basis des Raums W_j an:

Satz 2.5 (Mallat, Meyer) *Sei ϕ eine „Scalingfunction“ und ψ eine Funktion, deren Fouriertransformierte folgende Form hat:*

$$\hat{\psi}(\omega) = \frac{1}{\sqrt{2}} \hat{g}\left(\frac{\omega}{2}\right) \hat{\phi}\left(\frac{\omega}{2}\right) \tag{2.53}$$

mit

$$\hat{g}(\omega) = e^{-i\omega} \hat{h}^*(\omega + \psi) \tag{2.54}$$

und

$$\hat{h}(\omega) = \sqrt{2} \frac{\hat{\phi}(2\omega)}{\hat{\phi}(\omega)}. \tag{2.55}$$

Dann bildet für alle Skalen 2^j die Funktionenfamilie

$$\left\{ \psi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{t - 2^j n}{2^j}\right) \right\}_{n \in \mathbb{Z}} \tag{2.56}$$

eine orthonormale Basis für W_j. Für alle Skalen ist $\{\psi_{j,n}\}_{(j,n)\in\mathbb{Z}^2}$ eine orthonormale Basis des $L^2(\mathbb{R})$.

Ein Beweis dieses Satzes findet sich in [Mall98, S. 237ff]. Die soeben konstruierten $\psi_{j,n}$ sind die gesuchten orthonormalen Wavelets.

Beispiel 2.5 (Das Haar-Wavelet) Ein Beispiel für ein Wavelet, das eine orthonormale Basis des $L^2(\mathbb{R})$ bildet ist das Haar-Wavelet (siehe Abb. 2.5). Seine einfache Form erlaubt eine unkomplizierte Überprüfung der meisten der geforderten Eigenschaften an ein diskretes Wavelet[10]. Die mathematische Definition des Haar-Wavelets lautet:

$$\psi(t) = \begin{cases} 0 & , \quad t < -\frac{1}{2} \\ -1 & , \quad -\frac{1}{2} \leq t < 0 \\ 1 & , \quad 0 \leq t < \frac{1}{2} \\ 0 & , \quad \frac{1}{2} \leq t \end{cases} \tag{2.57}$$

Die zugehörige „Scalingfunction“ ist:

$$\phi(t) \begin{cases} 0 & , \quad t < -\frac{1}{2} \\ 1 & , \quad -\frac{1}{2} \leq t < \frac{1}{2} \\ 0 & , \quad \frac{1}{2} \leq t \end{cases} \tag{2.58}$$

Besonders die Modellierung digitaler Daten und anderer Treppenfunktionen bietet sich durch die Form des Wavelets an. Der unstetige Verlauf des Wavelets schränkt seine Eignung für die Analyse kontinuierlicher Signale oder anderer stetiger Funktionen ein.

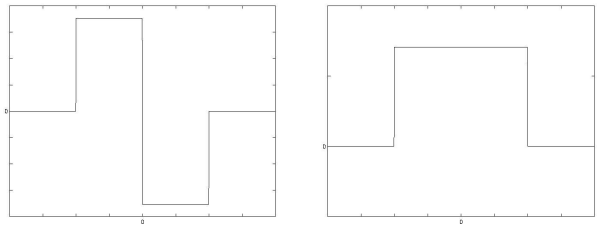


Abbildung 2.5: Das Haarwavelet und die zugehörige „Scalingfunction“

Multiskalenanalyse (MSA)

Orthonormale Waveletbasen bilden die Grundlage für die effiziente, rekursive Multiskalenanalyse.

Bemerkung 2.7 (Einschränkung)
In diesem Abschnitt soll nur ein knapper Einblick in die MSA unter der Verwendung orthonormaler Waveletbasen gegeben werden. Für genauere Details und die exakten mathematischen Zusammenhänge sei der interessierte Leser auf die entsprechenden Abschnitte in [Mall98] oder [Schr00] verwiesen.

[10] Eine derartige Überprüfung wird an dieser Stelle jedoch nicht vorgeführt.

In jedem Schritt i der Multiskalenanalyse werden von dem untersuchten Signal die Details der Stufe i separiert. Als Resultat erhält man eine Approximation $A_i f$ des Signals und eine Darstellung der Details $D_i f$. Dabei gilt [Schr00, S. 40ff]:

$$A_i f = \left(\left(\sqrt{2^i} f(u) \star \phi_{2^i}(-u) \right) (2^i n) \right)_{n \in \mathbb{Z}} \tag{2.59}$$

$$D_i f = \left(\left(2^i f(u) \star \psi_{2^i}(-u) \right) (2^i n) \right)_{n \in \mathbb{Z}} . \tag{2.60}$$

Dieses kann als eine Abbildung des Signals $f(t)$ in die Räume V_i bzw. W_i verstanden werden.

Mit wachsendem Index i wird die Approximation des Signals stetig gröber. Diese Art des Vorgehens erlaubt effiziente rekursive Implementierungen indem der Folgeschritt der Transformation $i+1$ nicht mehr auf den originalen Signaldaten, sondern auf der Approximation A_i arbeitet.

Die Darstellung ist vollständig. Ein Signal $f(t)$ kann vollständig aus seiner Approximation und den Details wiedergewonnen werden (wobei in der folgenden Definition $A_1 f$ dem ursprünglichen Signal entspricht):

Definition 2.12 (Orthogonale Waveletrepräsentation) *Das Ausgangssignal $A_1 f$ lässt sich für jedes $J > 0$ in der Form*

$$(A_J f, (D_i f)_{1 \leq i \leq J}) \tag{2.61}$$

darstellen. Diese Darstellung heißt orthogonale Waveletrepäsentation des Signals $A_i f$ der Tiefe J und besteht aus der Approximation $A_J f$ und den Detailinformationen der Skalierungen i für $1 \leq i \leq J$.

Die genauen mathematischen Zusammenhänge werden in [Schr00, S. 34ff] erläutert.

In der Praxis werden zur Durchführung der MSA zwei Filter eingesetzt, die sich aus den Wavelets bzw. der „Scalingfunction“ ableiten. Ein Tiefpassfilter H, aus der „Scalingfunction“ gewonnen, überführt die Approximation $A_i f$ in $A_{i+1} f$. Ein Hochpassfilter G ermittelt die, in diesem Approximationschritt verlorengegangenen Detailinformationen $D_i f$.

„Wavelet Packets“

Eine mit der MSA verwandte Art der waveletbasierten Analyse von Signaldaten ist die Verwendung von *„Wavelet Packets“* [Mall98, S. 319ff]: Anders als bei der MSA, die orthogonale Wavelets verwenden und pro Auflösungs-

schritt jeweils eine Approximation und eine Darstellung der Details der betreffenden Auflösung generieren, um dann im nächst gröberen Schritt lediglich mit der Approximation weiter zu verfahren (siehe oben), findet bei der Analyse mit „Wavelet Packets“ eine Weiterverarbeitung von sowohl der Approximation, als auch der Detailinformationen statt.

Die mathematische Grundlange dieses Vorgehens ist eine immer feiner werdende orthogonale Teilung der Basis des Signalraumes. Wie bei der MSA findet im ersten Auflösungsschritt eine orthogonale Teilung der Basis in die Subräume der Approximation und der Detailinformationen statt. In den nächsten Schritten werden diese Basen jeweils wieder in zwei orthogonale Subbasen aufgeteilt und die Signalinformationen bezüglich dieser Basen aufgeteilt.

Die Analyse eines Signals S geschieht also in der Form eines binären Baumes: Im ersten Schritt wir das Signal S in die Approximation der erste Stufe (S_1^0) und die Details der ersten Stufe (S_1^1) geteilt. Im nächsten Schritt werden sowohl S_1^0 wie auch S_1^1 in Approximation und Details zerlegt, resultierend in den Teildarstellungen S_2^0, S_2^1, S_2^2 und S_2^3, wobei der hochgestellte Index für die Auflösungsstufe steht und der tiefgestellte Index als Feinheit der Details interpretiert werden kann.

2.1.8 Zweidimensionale Ansätze

Um eine angemessene Möglichkeit zur Untersuchung zweidimensionaler Daten bzw. Signale, wie es z. B. Bilddaten sind, zu schaffen, existieren verschiedene Ansätze, die Waveletanalyse auf den zweidimensionalen Raum zu erweitern. Die Basis für die im folgenden vorgestellten Waveletanalysen bilden Familien von zweidimensionalen Wavelets $\psi(x, y)$. Die Art der Konstruktion dieser Wavelets ist für jeden der vorgestellten Ansätze verschieden.

Der Raum der zu analysierenden Daten ist im folgenden nicht mehr der $L^2(\mathbb{R})$ sondern der $L^2(\mathbb{R}^2)$. Der $L^2(\mathbb{R}^2)$ lässt sich über das *Tensorprodukt* aus Elementen des $L^2(\mathbb{R})$ bilden:

Seien $u, v \in L^2(\mathbb{R})$, dann ist $u \otimes v \in L^2(\mathbb{R}^2)$, wobei $u \otimes v$ wie folgt definiert ist:

$$\begin{array}{lcll} (u \otimes v) : \mathbb{R}^2 & \rightarrow & \mathbb{R} & \\ (u \otimes v)(x, y) & \mapsto & u(x)v(y) & \forall x, y \in \mathbb{R} \end{array}$$

Somit entspricht der Raum $L^2(\mathbb{R}^2)$ der Menge aller reellen endlichen Linearkombinationen

$$f(x, y) = \lambda_1 u_1(x)v_1(y) + \ldots + \lambda_n u_n(x)v_n(y) \tag{2.62}$$

für alle $\lambda_n \in \mathbb{R}$.

Ein zweidimensionales Wavelet verfügt über je zwei Skalierungs- und zwei Translationsparameter. Für eine zweidimensionale Funktion $f(x,y) \in L^2(\mathbb{R}^2)$ und ein zweidimensionales Wavelet $\psi(x,y)$ ergibt sich analog zu Definition 2.8 die zweidimensionale Wavelettransformation wie folgt:

$$Wf(u_x, s_x, u_y, s_y) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)\psi_{u_x,s_x,u_y,s_y}(x,y)\,dx\,dy. \tag{2.63}$$

Separable zweidimensionale Wavelets

Zur Verallgemeinerung der Wavelettheorie auf zwei Dimensionen, erweitert [Mall98, S. 304ff] die Methoden der diskreten Wavelettransformation. Wie im eindimensionalen Fall findet eine fortschreitende Aufspaltung und Verfeinerung der Basis des betrachteten Raums statt. Ausgehend von einer Basis diskreter Wavelets $\psi(t)$ mit zugehöriger „Scalingfunction“ $\phi(t)$ des $L^2(\mathbb{R})$ kann eine diskrete Basis des $L^2(\mathbb{R}^2)$ analog zu der Methode im eindimensionalen Fall konstruiert werden:

Die zweidimensionale „Scalingfunction“

$$\phi_{n,m}(x,y) = \phi_n(x)\phi_m(y) \tag{2.64}$$

bildet die Basis des Raums $V_0^2 \subset L^2(\mathbb{R}^2)$. Wie in Beispiel 2.4 kann von diesem Raum ausgehend durch Hinzufügen von Detailräumen W_j^2 jeder feinere Raum V_j^2 konstruiert werden:

$$V_{j-1}^2 = V_j^2 \oplus W_j^2 \tag{2.65}$$

Die Basis des Detailräume W_j^2 bildet sich aus drei Wavelets (siehe Abb. 2.6):

$$\psi_{j,n,m}^1(x,y) = \phi_{j,n}(x)\psi_{j,m}(y) \tag{2.66}$$
$$\psi_{j,n,m}^2(x,y) = \psi_{j,n}(x)\phi_{j,m}(y) \tag{2.67}$$
$$\psi_{j,n,m}^3(x,y) = \psi_{j,n}(x)\psi_{j,m}(y) \tag{2.68}$$

Als Ergebnis eines Transformationschritts eines Signals $f(x,y)$ unter Verwendung dieses Ansatzes ergeben sich somit eine Approximation und drei Detailansichten. Für die exakten mathematischen Hintergründe sei an dieser Stelle auf die Werke [Mall98] und [Schr00] verwiesen.

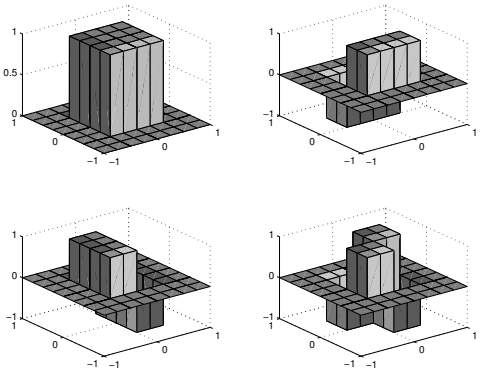


Abbildung 2.6: Zweidimensionale Haar-„Scalingfinction“ und Haar-Wavelets $(\phi, \psi_1, \psi_2, \psi_3)$

Zweidimensionale Wavelets zur Bildverarbeitung

W. Bäni stellt in [Bäni01] zwei Ansätze für zweidimensionale Wavelets vor. Der theoretisch gehaltene Abschnitt über separable zweidimensionale Wavelets [Bäni01, Seite 129ff] entspricht der Methode von Mallat (siehe oben). Später im Buch stellt er einen weiteren Ansatz vor, der aus praktischen Erfahrungen in der Bildverarbeitung resultiert:

Für den vorgestellten Ansatz, der seinen Ursprung in der Kantenfindung hat, verwendet Bäni Wavelets, die sich aus der ersten Ableitung einer Glättungsfunktion ergeben. Im Falle des Haar-Wavelets (siehe Beispiel 2.5) ist die betreffende Glättungsfunktion:

$$\varphi(t) = \begin{cases} 0 & , \quad t < -\frac{1}{2} \\ \frac{1}{2} + t & , \quad -\frac{1}{2} \leq t < 0 \\ \frac{1}{2} - t & , \quad 0 \leq t < \frac{1}{2} \\ 0 & , \quad \frac{1}{2} \leq t \end{cases} \tag{2.69}$$

Für das Haar-Wavelet gilt: $\psi(t) = \varphi'(t)$. Aus dem Wavelet und der Glättungsfunktion bildet Bäni zwei separable zweidimensionale Wavelets (siehe Abb. 2.7):

$$\psi^x(x, y) = \psi(x)\varphi(y) \tag{2.70}$$

$$\psi^y(x, y) = \varphi(x)\psi(y) \tag{2.71}$$

Im weiteren Verlauf des entsprechenden Abschnittes (siehe [Bäni01, S. 226ff]) wird dieser Ansatz auf nicht seperable, zweidimensionale Wavelets verallgemeinert.

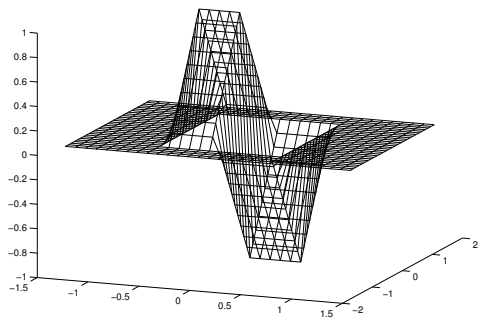


Abbildung 2.7: Zweidimensionales Haarwavelet nach [Bäni01]

Zweidimensionale Gaborwavelets

Die Konstruktion zweidimensionaler Gaborwavelets, wie sie z. B. der Algorithmus von John Daugman verwendet, verhält sich analog zum eindimensionalen Fall: Ein zweidimensionales Gauß-Fenster $g(x, y)$, das von einer komplexen Welle der Frequenz ξ moduliert wird (siehe Abb. 2.8):

$$\psi(x, y) = g(x, y)\, e^{i\xi x} \tag{2.72}$$

Wie dieser Formel entnommen werden kann, findet eine Frequnez-Modulierung des Gauß lediglich in Richtung der x-Achse statt. Aus diesem Grund tragen zweidimensionale Gaborwavelets in den meisten Fällen einen weiteren Parameter, der für die räumliche Ausrichtung des Wavelets verantwortlich ist. Die genaue Anwendung dieses Richtungsparameters unterscheidet sich je nach gesichteter Quelle. [Hama99] verwendet folgende Variante:

$$\psi(x, y, \vartheta) = e^{-\frac{1}{2}\left(\frac{(x\cos\vartheta + y\sin\vartheta)^2}{\sigma_x^2} + \frac{(-x\sin\vartheta + y\cos\vartheta)^2}{\sigma_y^2}\right)} \cdot e^{\frac{2\pi(x\cos\vartheta + y\sin\vartheta)}{\xi} i} \tag{2.73}$$

Zur Integration der Richtungsinformationen wurde somit hier die Standard-Rotationsmatrix auf die Koordinaten (x, y) angewendet (siehe hierzu auch Abschnitt 3.2.1):

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos\vartheta_k & \sin\vartheta_k \\ -\sin\vartheta_k & \cos\vartheta_k \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} \tag{2.74}$$

Eine Reihe von Anwendungen zweidimensionaler Gaborwavelets folgt in Kapitel 3.

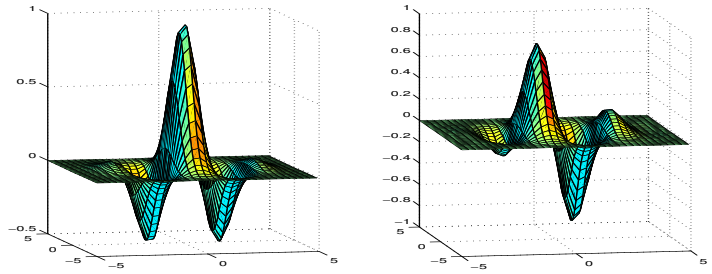


Abbildung 2.8: Zweidimensionales Gaborwavelet (Real- und Imaginärteil)

2.1.9 Anwendungen von Wavelets

Im Laufe der Zeit hat sich eine große Bandbreite an Anwendungen der Wavelettheorie entwickelt. Kapitel 3 beschreibt verschiedene Verwendungsmöglichkeiten in der biometrischen Authentikation. Aus Platzgründen an dieser Stelle nur drei weiterführende Beispiele:

- Datenkompression

  Die Anwendung von Wavelets zur Datenkompression basiert auf der Beobachtung, dass bei einer Rekonstruktion der Daten aus einer Wavelettransformation, das Auslassen der Waveletkoeffizienten kleinen Betrags kaum signifikante Unterschiede zwischen den Originaldaten und der Rekonstruktion verursacht. Bei der Anwendung dieses Ansatzes der Datenkompression werden, im Anschluss an eine Wavelettransformation der Daten, alle Koeffizienten verworfen, die unterhalb eines festgelegten Schwellwertes liegen. Die verbleibenden Koeffizienten, deren Anzahl häufig um einige Größenordnungen kleiner ist, kann effizient gespeichert werden. Diese Art der Datenkompression ist verlustbehaftet.

  Zwei Beispiele: Der Kompressionsstandard JPG200 verwendet Wavelets und die Fingerabdruckdatenbank des FBIs benutzt Wavelets zur Kompression der Daten.

- Entrauschung

  Bei dem Vorgang der Entrauschung werden Waveletkoeffizienten verworfen deren Größe unterhalb einer festgelegten Grenze liegt. Verrauschung von Signalen schlägt sich im Allgemeinen in Störungen in den feinen Details der Signaldaten wieder. Je kleiner das Wavelet, desto feiner sind die Detailinformationen des Signals, die in den betreffenden Koeffizienten kodiert werden. Beim Verwerfen dieser Informationen

findet eine Betonung der gröberen Verläufe des Signals statt, bei gleichzeitiger Eliminierung der Störungen im feinen Detailraum. Je nach der vorliegenden Signalcharakteristik können so befriedigende Ergebnisse erziehlt werden.

So konnten durch Wavelettechniken historische, schon verlorene Tonaufnamen eines Klavierkonzerts soweit rekonstruiert werden, dass eine Feststellung der gespielten Noten möglich war [Hubb97].

- Kantenfindung

  Wie schon erwähnt, weisen bei den meisten Wavelets große Waveletkoeffizienten auf starke Änderungen in den Signaldaten hin. Dieses kann zum Zweck der Kantendetektion ausgenutzt werden.

2.2 Authentikation

Eine wichtige Grundlage von Verfahren zur Durchsetzung von Sicherheitsregeln ist die Zugriffskontrolle[11]. Diese regelt die Zugriffsrechte von Personen auf Ressourcen (dieses können beispielsweise sowohl IT-Systemressourcen, wie Daten oder Applikationen, als auch physikalische Ressourcen, wie der Zugang zum Serverraum, sein).

Voraussetzung für eine effektive Zugriffskontrolle ist eine zuverlässige Verifikation der Identität von Personen. Methoden zur Durchführung dieser Aufgabenstellung werden unter dem Begriff der *Authentikation*[12] zusammengefasst [Goll99]:

Definition 2.13 *Unter dem Begriff der Authentikation versteht man, den Vorgang der Verifikation einer angenommenen oder behaupteten Identität.*

Vor der Authentikation muss in den meisten Fällen eine Festlegung der zu verifizierenden Identität erfolgen[13]. Dieses erfolgt üblicherweise durch die Eingabe einer eindeutigen Benutzerkennung.

Ist der Träger der angegebenen Identität berechtigt, auf die kontrollierten Ressourcen zuzugreifen, muss eine Verifikation dieser Identität erfolgen, um Missbrauch durch fälschlich angenommene Identitäten auszuschließen. Je

[11] Auch wenn die folgenden Beschreibungen stark in der IT-Welt fußen, beschränkt sich die Anwendung der hier eingeführten Methoden zur Personen-Authentikation nicht auf den Bereich der IT-Sicherheit. Dieselben Methoden finden sich in vielen anderen Bereichen des modernen Lebens, wie z. B. bei der Benutzung eines Geldautomaten.

[12] In der Literatur finden sich neben den Begriff der Authentikation auch die Bezeichnungen „Authentisierung“ oder „Authentifikation“, die ebenfalls den hier erläuterten Begriff beschreiben.

[13] Eine Ausnahme bildet hier u.a. die biometrische Identifikation (siehe Abschnitt 2.18).

nach Anwendungsszenario können eine oder mehrere der folgenden Methoden zum Einsatz kommen.

Um die Notation einfach zu halten, wird im folgenden stets von einem *System*, das Ressourcen zur Verfügung stellt, und einem *Nutzer*, der auf die Systemressourcen zugreifen möchte, die Rede sein. Der Nutzer ist in diesem Fall die Instanz, die ihre Identität beweisen muss. Das System ist die Instanz, welche die Identität verifiziert. Diese Darstellung ist vereinfacht. Andere Authentikationsszenarien ergeben sich analog.

2.2.1 Authentikation durch Wissen

Diese weit verbreitete Methode der Authentikation weist jeder Benutzeridentität ein Geheimnis zu, das mit der authentifizierenden Instanz geteilt wird. Im Anschluss an die Feststellung der Identität beweist der Nutzer dem System, dass er Kenntnis über das geteilte Geheimnis besitzt. Dieses kann z. B. über die Eingabe eines Passwortes oder einer PIN („Personal Identification Number“) geschehen.

Die Identität des Nutzers gilt als bewiesen, wenn er dem System gegenüber Kenntnis des Geheimnisses beweisen konnte.

Die Sicherheit dieser Methode hängt stark von dem Benutzer ab. Wenn dieser es versäumt, das Geheimnis ausreichend effektiv zu schützen, kann die Zuverlässigkeit der Authentikation kompromittiert werden. Eine weitere Schwäche dieses Verfahrens (besonders in dem Fall der Verwendung von Passwörtern) ist eine schlechte Wahl des Geheimnisses. Dieses betrifft sowohl zu einfache Geheimnisse, die leicht erraten werden können, als auch zu komplizierte Geheimnisse, deren Verwendung unsichere Hilfsmittel erfordert (wie z. B. lange zufällige Zeichenfolgen als Passwort, die durch ihre schwere Merkbarkeit den Nutzer dazu bringen, das Passwort zu notieren).

2.2.2 Authentikation durch Besitz

Bei der Authentikation über Besitz weist Nutzer seine Identität nach, indem er dem System beweist, dass er im Besitz eines eindeutigen Gegenstandes ist. Dieser Gegenstand kann z. B. ein Dongle, eine Magnet- oder Chipkarte sein. Da der Gegenstand dem Nutzer eindeutig zugeordnet ist, findet im Allgemeinen auch die Feststellung der zu prüfenden Identität über den Gegenstand statt.

Diese Methode wird häufig in Zusammenarbeit mit anderen Authentikationsmethoden, wie z. B. über Wissen, eingesetzt, da ein Diebstahl des Gegenstandes nicht ausgeschlossen werden kann. Ein Beispiel für eine derartige Kombination der Methoden ist die bereits erwähnte Bankkarte: Um auf sein

Konto zugreifen zu können, führt der Nutzer zuerst seine persönliche Bankkarte in den Geldautomaten ein. Ein Zugriff auf das Konto ist jedoch erst nach korrekter Eingabe der Geheimzahl des Nutzers möglich.

2.2.3 Authentikation durch Ort und Zeit

Ebenfalls üblicherweise in Kombination mit anderen Methoden der Authentikation eingesetzt, verlangt die Authentikation durch Ort und Zeit vom Nutzer, dass die Authentikation zu einem bestimmten Zeitpunkt bzw. an einen bestimmten Ort durchgeführt wird.

So kann festgelegt werden, dass auf bestimmte Systemressourcen nur von Rechnern innerhalb des Firmengeländes oder während der Geschäftszeiten zugegriffen werden kann.

2.2.4 Authentikation durch biometrische Merkmale

Bei dieser Art der Authentikation weist der Nutzer seine Identität darüber nach, dass er Träger eines spezifischen biometrischen Merkmals ist[14]. Diese Art der Authentikation wird ausführlich im Abschnitt 2.3 erläutert.

2.2.5 Authentikationsmodelle

In der aktuellen Ausprägung werden zwei Modelle der Authentikation unterschieden: „*Single Sign On*“ und „*Multiple Sign On*“.

Der Begriff des „Multiple Sign On“ beschreibt den Zustand, dass die Authentikation ressourcenabhängig vollzogen wird. Jede der zugriffsbeschränkten Ressourcen verfügt über eine eigene Instanz, welche die Verifikation der behaupteten Identität übernimmt. Der Nutzer authentisiert sich somit gegenüber der Ressource direkt. Dieser Ansatz beinhaltet die mehrfache Haltung von Identitätsinformationen der Nutzer. Somit ist es möglich, dass dieselbe Person auf unterschiedliche Ressourcen mit verschiedenen „Identitäten“ , die in keiner Relation zueinander stehen, zugreift. Eine pseudonyme Nutzung dieser Ressourcen ist somit realisierbar. Der Nachteil dieses Modells ist, dass der Nutzer jeder einzelnen Ressource, auf die er zugreifen möchte, separat bekannt gemacht werden muss. Folglich muss der Nutzer für jede dieser Ressourcen spezielle Mittel der Authentikation bereitstellen (verschiedene Passwörter, Chipkarten, etc.). Daraus erwachsen in nicht wenigen Fällen Sicherheitsprobleme (z. B. durch mehrfach verwendete Passwörter).

[14] Genaugenommen ist diese Art der Authentikation ein Sonderfall der „Authentikation über Merkmale“. Diese Einschränkung sollte bedacht werden.

Unter dem Begriff des „Single Sign On“ versammeln sich zentralisierte Authentikationsanwendungen. Alle Ressourcen, die sich unter einer dieser Anwendungen sammeln, verwenden dieselbe Instanz zur Verifikation der behaupteten Identitäten der Nutzer, unabhängig von der sonstigen Verzahnung dieser Ressourcen. Der Nutzer muss somit für den Zugriff auf eine Reihe von Ressourcen lediglich ein Mittel der Authentikation zur Verfügung stellen. Je nach Ausprägung kann dieser Ansatz eine pseudonyme Nutzung erschweren.

2.3 Grundlagen der biometrischen Methoden zur Authentikation

2.3.1 Grundlegende Begriffe

Der Begriff *Biometrik* setzt sich aus den Bestandteilen *Biologie* und *Metrik* (siehe Anhang B.4) zusammen. Das Feld der Biometrik umfasst somit Verfahren und Methoden, die es ermöglichen, ausgewählte biologische Eigenschaften des Menschen in einem Raum von Kenngrößen abzubilden, die einen Vergleich dieser Merkmale mittels mathematischen Methoden erlauben.

Definition 2.14 (Biometrischer Algorithmus, biometrische Signatur)
Ein biometrischer Algorithmus ist ein Verfahren, das aus einem biologischen Merkmal eine vergleichbare Kenngröße generiert. Diese Kenngröße trägt die Bezeichnung biometrische Signatur.

Zentral für einen biometrischen Algorithmus ist das von ihm erfasste *biometrische Merkmal*. Ein biometrisches Merkmal ist eine messbare physiologische oder verhaltensbasierte Charakteristik eines Lebewesens [Zhan00]. Damit ein Merkmal für die biometrische Authentikation geeignet ist, muss es folgende Eigenschaften besitzen:

- Universalität, Verbreitung:

  Jedes Mitglied der betrachteten Personengruppe sollte das verwendete Merkmal besitzen. Bei einem biometrischen Algorithmus, der auf universelle Einsetzbarkeit abzielt, sollte das Merkmal demzufolge in der gesamten Menschheit auftreten. Ist dies bei den meisten physiologischen Merkmalen noch der Fall, treten bei einigen verhaltensbasierten Charakteristika an diesem Punkt Begrenzungen auf. So ist z. B. der in [Zhu00b] beschriebene Algorithmus, der eine Methode zur Personenidentifikation über Handschriftenanalyse beschreibt, nur für Bevölkerunggruppen, denen die Schreibweise chinesischer Schriftzeichen geläufig ist, anwendbar.

- Einzigartigkeit:

  Das verwendete Merkmal sollte für jedes Individuum hinreichend verschieden ausgeprägt sein. Diese Eigenschaft ist grundlegend, um eine Unterscheidung und Erkennung von Personen anhand dieses Merkmals zu ermöglichen.

- Persistenz:

  Das Merkmal sollte sich innerhalb eines definierten Zeitraum nicht signifikant ändern. Je nach Algorithmus und Anwendung können unterschiedliche Ansprüche an die Persistenz eines Merkmals gestellt werden. Verbreitet ist die Anforderung, dass die Ausprägung des Merkmals über eine größeren Zeitraum des menschlichen Lebens nahezu konstant bleibt.

In der Literatur trifft man weiterhin auf die Anforderung der „Erfassbarkeit“ des Merkmals, also der Anforderung der Möglichkeit zur Aufnahme des Merkmals mittels vorhandener Technik. Auch wenn diese Eigenschaft zweifelsfrei eine Voraussetzung für die Verwendung des Merkmals im Rahmen der Biometrik ist, wurde sie nicht in die obenstehende Auflistung integriert, da es sich anders als bei den oben genannten Kriterien, nicht um eine Eigenschaft des Merkmals, sondern um eine Eigenschaft der verwendeten Technik handelt. Somit unterliegt die Anforderung der Erfassbarkeit anderen Bedingungen als die oben aufgeführten. Wenn ein Merkmal heute als nicht erfassbar gilt, kann sich dieses durch technische Fortschritte im kommenden Jahrzehnt durchaus ändern.

Je nach dem Faktor, der die individuelle Ausprägung eines biometrischen Merkmals bestimmt, findet eine Klassifizierung der Merkmale statt[15]:

- Genotypisch: Die Ausprägung des Merkmals ist rein durch die Gene des Menschen bestimmt.

  Beispiel: DNA.

- Phenotypisch: Umwelteinflüsse tragen zu der Ausprägung des Merkmals bei oder bestimmen diese vollständig.

  Beispiel: Fingerabdruck, Iris, Retina, Gesicht.

- Konditioniert: Das Merkmal ist verhaltensbasiert. Die Ausprägung wird durch erlernte Abläufe bestimmt.

  Beispiele: Gang, Handschrift, Anschlagsdynamik.

[15] In vielen Fällen ist die Bestimmung, welcher der folgenden drei Klassen ein Merkmal angehört nicht abgeschlossen.

Diesen Abschnitt abschließen soll der Begriff der *Lebenderkennung*:

> „Unter Lebenderkennung wird die Überprüfung verstanden, ob die jeweiligen biometrischen Merkmale auch von einem lebenden Organismus und nicht von einer künstlichen Fälschung erzeugt wurden.“ [Wirt99]

Die Lebenderkennung ist kein integraler Bestandteil eines biometrischen Algorithmus. Sie ist für einen Einsatz der Biometrik zum Zwecke der Authentikation jedoch unerlässlich.

2.3.2 Biometrische Prozesse

Dieser Abschnitt beschäftigt sich mit den Prozessen, die im Rahmen des Betriebs einer biometrischen Anwendung ablaufen. Diese umfassen sowohl administrative (z. B. das Einlernen von Personen), wie auch operative Prozesse (z. B. die biometrische Authentikation).

Bemerkung 2.8 (Biometrische Datenbank)
Im folgenden wird der Begriff *biometrische Datenbank* verwendet. Dieser Begriff wird im Rahmen dieser Arbeit nicht vollständig definiert[16]. Eine biometrische Datenbank, so die weitere Verwendungsweise des Begriffs, speichert eine Reihe von Zwei-Tupeln (P, S), wobei P für eine eindeutige Personenidentifikation steht und S die biometrischen Daten repräsentiert, die zu der Person P gehören. S kann je nach verwendetem Algorithmus (siehe hierzu auch Kapitel 3) eine einzelne biometrischen Signatur oder ein Satz von biometrischen Daten enthalten. Die Datenbank erlaubt so eine Zuordnung von biometrischen Daten S zu Personendaten P. Vergleichsmethoden stellt sie jedoch nicht zur Verfügung.

Weiterhin wird im folgenden von *biometrischen Systemen* gesprochen. Ein biometrisches System ist, im Rahmen dieser Arbeit, die Vereinigung eines biometrischen Algorithmus, einer biometrischen Datenbank und einer Implementierung der biometrischen Prozesse.

Administrative Prozesse

Im Rahmen der Administration eines biometrischen Systems ist es notwendig, Personen dem System hinzuzufügen bzw. Personen aus dem System zu entfernen.

[16] In diesem Falle ist es notwendig, neben den beschriebenen Eigenschaften, auch Aspekte wie Datensicherheit, Schutz von personengebundenen Daten, sichere Übertragung oder Speicherung der biometrischen Rohdaten zu diskutieren.

Definition 2.15 (Einlernen („Enrollment“)) *Beim Einlernen einer Person in ein biometrisches System werden für diese Person Referenz-Signaturen generiert und in der biometrischen Datenbank abgelegt.*

Die Anzahl und Eigenschaften der Referenz-Signaturen ist abhängig von dem eingesetzten Algorithmus.

Definition 2.16 (Entfernen („Derollment“)) *Beim Entfernen einer Person aus einem biometrischen System werden sämtliche auf diese Person bezogene Datensätze aus der Datenbank gelöscht.*

Bemerkung 2.9 (Aktualisierung von Datensätzen)
Eine ebenfalls notwendige administrative Tätigkeit ist das Aktualisieren von Daten bereits eingelernter Personen. Der Einfachheit halber soll diese Tätigkeit hier lediglich als die Kombination aus einem Entfernen der betreffenden Person aus der Datenbank und einem anschließenden neuen Einlernen aufgefasst werden.

Operative Prozesse

Die in dieser Arbeit betrachtete Anwendung der Biometrik ist die Authentikation durch biometrische Eigenschaften (siehe Abschnitt 2.2.4), im folgenden *biometrische Authentikation* genannt.

Bei der biometrischen Authentikation können zwei Prozesse unterschieden werden: die *Verifikation* und die *Identifikation*. Den beiden Prozessen ist gemein, dass sie im Anschluss an eine Berechnung von biometrischen Daten auf die biometrische Datenbank zugreifen, um die erfasste Signatur mit den eingelernten Datensätzen zu vergleichen.

Definition 2.17 (biometrische Verifikation) *Bei der biometrischen Verifikation wird eine angegebene Personenkennung anhand von erfassten biometrischen Daten überprüft.*

Zu diesem Zweck findet ein 1:1-Zugriff auf eine biometrische Datenbank statt. Die erfassten Daten werden allein mit den der angegebenen Personenkennung zugeordneten biometrischen Daten verglichen.

Sind sich die erfassten und die gespeicherten Daten hinreichend ähnlich, terminiert die biometrische Verifikation erfolgreich. Die behauptete Identität der Person konnte in diesem Fall bestätigt werden.

Beispiel 2.6 (Verifikation: Bankautomat)
Ein naheliegendes Beispiel für eine Anwendung der biometrischen Verifikation ist eine Integration biometrischer Techniken in Bankautomaten. Beim Zugriff auf sein Konto über einen Bankautomaten, gibt der Kunde seine

Identität an (z. B. über das Einführen seiner Bankkarte) und beweist der Maschine gegenüber seine Identität durch ein biometrisches Merkmal (und nicht wie z.Zt üblich über die Eingabe seiner Geheimnummer). Der Bankautomat erfasst das biometrische Merkmal des Kunden und führt eine biometrische Verifikation anhand der angegebenen Identität durch. Ist die biometrische Signatur des soeben aufgenommenen Merkmals der gespeicherten Referenzsignatur hinreichend ähnlich, erhält der Kunde Zugriff auf sein Konto.

Definition 2.18 (biometrische Identifikation) *Bei der biometrischen Identifikation wird überprüft, ob es eine Übereinstimmung zwischen den erfassten biometrischen Daten mit den in der Datenbank abgelegten Daten gibt. Dieses dient dem Zweck, Personen anhand ihrer biometrischen Merkmale zu identifizieren.*

Zu diesem Zweck findet ein 1:n-Zugriff auf die biometrische Datenbank statt. Die erfassten Daten werden mit sämtlichen gespeicherten Datensätzen verglichen.

Findet sich ein Datensatz in der biometrischen Datenbank, der den erfassten Daten hinreichend ähnlich ist, terminiert die Identifikation erfolgreich. In diesem Fall gibt die biometrischen Identifikation die Personenkennung zurück[17]*.*

Beispiel 2.7 (Identifikation: Das Sportstadion)
Eine denkbare Anwendung der biometrischen Identifikation ist eine Überwachung öffentlicher Räume auf die Anwesenheit vorgegebener Personen. So können z. B. die Ränge eines Sportstadions darauf überprüft werden, ob sich polizeibekannte Hooligans unter der Zuschauern befinden. Zu diesem Zweck werden die Zuschauertribünen des Stadions mit Digitalkameras abgefilmt und das so produzierte Bildmaterial in eine biometrische Anwendung gegeben, die aus den Bildern die Gesichter der Zuschauer isoliert und diese mittels Verfahren der Gesichtserkennung klassifiziert. Die so gewonnenen biometrischen Signaturen werden mit einer Datenbank verglichen, die Signaturen der Gesichter zuvor auffällig gewordener Hooligans enthält. Kommt eine der aus den Zuschauerdaten gewonnene Signatur einer eingelernten Signatur der Datenbank hinreichend nahe, ist einer der Zuschauer als bekannter Hooligan identifiziert worden[18].

Weiterhin existieren Anwendungen, deren Zielsetzung weniger eng gefasst ist als die oben beschriebenen Verifikation und Identifikation. Den oben angeführten Anwendungen ist es gemein, dass am Ende des Prozesses genau ein

[17] Das Verhalten einer biometrischen Identifikation, bei der mehr als ein Datensatz als hinreichend ähnlich zu den erfassten Daten erkannt wird, ist undefiniert und hängt von der jeweiligen Implementierung ab.

[18] Diese Methode wurde z. B. Anfang 2003 in amerikanischen Stadien während des „Super Bowls“ eingesetzt (siehe http://www.wired.com/news/culture/0,1284,56878,00.html).

Resultat steht: Im Falle der Verifikation „erkannt“ bzw. „nicht erkannt“ . Im Falle der Identifikation wäre das Ergebnis entweder die ermittelte Identität des Trägers des untersuchten Merkmals oder die Meldung, dass keine erfolgreiche Identifikation vorgenommen werden konnte. In [Ashb02, S. 45] wird eine Anwendung im Bereich der Fingerabdruckerkennung beschrieben, die zur Unterstützung der klassischen forensischen Methoden aus einer großen Anzahl von Fingerabdrücken ein Subset von in Frage kommenden Exemplaren ermittelt. Dieses Subset wird anschließend von menschlichen Experten weiter untersucht. Am Ende dieser der Identifikation verwandten Methode steht also kein einzelnes Ergebnis, sondern eine Menge von Fingerabdrücken, die in einer definierten Ähnlichkeitsbeziehung zu dem untersuchten Abdruck stehen.

Ebenso sind Anwendungen vorstellbar, die nicht auf die Zuordnung von untersuchten biometrischen Merkmalen zu genau einer Person abzielen, sondern auf eine weitergehende Klassifizierung des Merkmalsträgers, wie z. B. ethnische Gruppenangehörigkeit, Geschlecht und ähnliches.

Diese Anwendungen sind mit denen der biometrischen Authentikation verwandt aber nicht identisch. Aus diesem Grund wird im weiteren Verlauf dieser Arbeit auf Anwendungen mit dieser Zielsetzung nicht eingegangen. Es soll aber darauf hingewiesen sein, dass aufgrund der Nähe dieser Anwendungen zur Identifikation/Verifikation die verwendeten Algorithmen den hier behandelten Methoden sehr ähnlich oder in Teilen mit ihnen identisch sein können.

2.3.3 Leistungskenngrößen

Zur Beurteilung der Qualität einer biometrischen Systems zur Authentikation dienen die Leistungskenngrößen *False Rejection Rate (FRR)* und *False Acceptance Rate (FAR)*. Beide Größen beziehen sich auf das Verhalten des Systems bei einem Einsatz zur Zugriffskontrolle.

False Rejection Rate (FRR)

Die FRR gibt den Anteil der autorisierten Nutzer an, die zu unrecht von dem System zurückgewiesen wurden. Die Kenngröße errechnet sich aus dem Verhältnis der absoluten Anzahl von Authentikationversuchen autorisierten Personen („number of authorized identification attempts“ (NAA)) zur Zahl der zu unrecht zurückgewiesenen Personen („number of false rejections“ (NFR)). Sie berechnet sich wie folgt [Zhan00]:

$$\mathrm{FRR} = \frac{\mathrm{NFR}}{\mathrm{NAA}} * 100\% \tag{2.75}$$

Je höher die FRR eines Systems ist, desto größer ist Anzahl der Personen, die zu unrecht als nicht autorisiert klassifiziert werden (was im Allgemeinen der Zahl der fehlerhaft nicht erkannten eingelernten Personen entspricht).

False Acceptance Rate (FAR)

Die FAR gibt an, wie hoch der Anteil der nicht autorisierten Personen ist, die durch fehlerhafte Erkennung dennoch Zugang zum System erhalten. Die FAR berechnet sich aus der Anzahl („number of false acceptances“ (NFA)) und der absoluten Anzahl der unautorisierten Zugriffsversuche („number of imposter identification attempts“ (NIA)). Sie berechnet sich wie folgt:

$$\mathrm{FAR} = \frac{\mathrm{NFA}}{\mathrm{NIA}} * 100\% \tag{2.76}$$

Je höher die FAR, desto größer ist die Wahrscheinlichkeit, dass ein Betrüger unter Annahme einer falschen Identität unautorisierten Zugriff auf das System erhält.

Equal Error Rate (EER)

Im Idealfall hat ein biometrisches System eine FAR und eine FRR von 0%. Da dieses im Allgemeinen nicht der Fall ist, müssen die Wechselwirkungen der beiden Größen betrachtet werden. Sowohl die FAR, als auch die FRR sind von der Kalibrierung des biometrischen Systems bestimmt. Die Parametrisierung des Schwellwerts, über den entschieden wird, ob die Ähnlichkeit einer berechneten Signatur zu einer Referenzsignatur groß genug ist, um einer eingelernten Person zugewiesen werden zu können, hat Einfluss auf die Kenngrößen. Ist der Schwellwert niedrig, muss die Ähnlichkeit der Signaturen also groß sein, führt dies zu einer niedrigen FAR, da die Annahme einer fremden Identität erschwert wird, und zu einer hohen FRR, da die Auswirkungen von Störungen in den Bilddaten des biometrischen Merkmals ein größeres Gewicht bekommen und so unter Umständen auch autorisierte Personen zurückgewiesen werden können.

Häufig wird eine Parametrisierung des Systems bestimmt, bei der sich die FAR und die FRR gleichen. Diesen Punkt bezeichnet man als die „equal error rate“ (EER) (siehe Abb. 2.9). Ein System mit einer Parametrisierung, die zu einer FAR von 3% und einer FRR von 3% führt, hat folglich eine EER von 3%. Diese Kenngröße dient dazu, die untere Grenze der Optimierung eines Systems zu bestimmen. Eine Veränderung der Parametrisierung, um eine der beiden Kenngrößen zu verbessern, hat zwangsläufig eine Verschlechterung der anderen Kenngröße zur Folge.

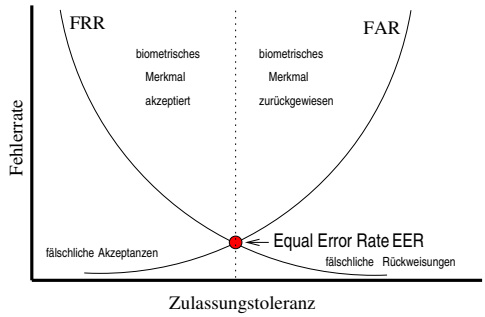


Abbildung 2.9: Bestimmung der EER [Froe03]

2.3.4 Ausgewählte biometrische Techniken

Im Laufe der Zeit haben sich zahlreiche biologische Merkmale des Menschen als geeignet für eine Verwendung in der biometrischen Authentikation erwiesen. Die Forschung in diesem vergleichsweise jungen Feld ist noch nicht abgeschlossen und mit einer Entwicklung weiterer, bisher unbekannter biometrischer Techniken ist zu rechnen. Auch im Bereich der etablierten biometrischen Techniken (wie z. B. die Fingerabdruckerkennung) ist die Forschung noch nicht zum Stillstand gekommen. Es wird sowohl an Verbesserngen bereits eingeführter Methoden, wie auch an der Entwicklung neuer Ansätze gearbeitet. Aus diesen Gründen hat der folgende Überblick über Techniken und Methoden keinerlei Anspruch auf Vollständigkeit, sondern soll lediglich als Sammlung von Beispielen dienen.

- Fingerabdruckerkennung

  Die Fingerabdruckerkennung ist der bekannteste Vertreter der biometrischen Techniken. Die Ursprünge dieser Technik gehen weit in die Zeit vor der Einführung des Computers zurück und haben ihre Wurzel in der polizeilichen Forensik. Verbreitete Verfahren bedienen sich zur Klassifizierung charakteristische Muster der Rillen des Fingerabdrucks, der sogenannten Minutien. Ein Algorithmus, der einen anderen, waveletbasierten Ansatz verfolgt, wird in Abschnitt 3.2.1 dieser Arbeit vorgestellt.

- Überprüfung der Handgeometrie

  Bei dieser Technik findet ein Vergleich von geometrischen Finger- oder Handcharakteristika - z. B. der Länge, Breite, Dicke der Finger und Eigenschaften der Handoberfläche - statt.

- Gesichtserkennung

  Für diese biometrische Technik existiert eine Reihe von verschiedenen Ansätzen (u.a. Vergleich über elastische Graphen, neuronale Netze oder mittels frequenzanalytischer Methoden). Zwei dieser Ansätze werden im Abschnitt 3.3 dieser Arbeit beschrieben.

- Iriserkennung

  Die Iriserkennung basiert auf der großen Variation der Irismuster des menschlichen Auges. Der Algorithmus von John Daugman zur Iriserkennung wird in Abschnitt 3.1.1 detailliert erläutert. In Kapitel 4 werden anhand dieses Algorithmus Methoden zur Implementierung und Parametrisierung waveletbasierter Algorithmen vorgeführt.

- Retina-Scan

  Die Retina des menschliche Auges (die Adernhaut im Inneren des Augapfels) besitzt eine, für jeden Menschen individuelle, Adernstruktur. Diese kann als Grundlage für eine biometrische Erkennung verwendet werden. Eine Aufnahme dieses Merkmals ist nicht ohne größeren technischen Aufwand zu vollziehen. Im Vergleich zu anderen biometrischen Techniken hat diese Methode den Vorteil, ist dass beim heutigen Stand der Technik eine unbemerkte Erfassung dieses biometrischen Merkmals praktisch nicht möglich ist (anders als z. B. bei Fingerabdrücken oder dem menschlichen Gesicht).

- Analyse der Tippdynamik

  Dieser biometrischen Technik liegt die Beobachtung zu Grunde, dass der Tipprhythmus eines Menschen unabhängig von der emotionalen Verfassung des Menschen und der verwendeten Tastatur ist (nach [Wirt99]).

- Sprechererkennung

  Aufnahmen von Sprechproben (je nach Algorithmus mit oder ohne vorgegebenen Text) werden auf personencharakteristische Eigenschaften untersucht.

Kapitel 3

Anwendungen von Wavelets in der biometrischen Authentikation

In den folgenden Abschnitten findet sich eine Übersicht verschiedener Anwendungen von Wavelets in der biometrischen Authentikation. Bei dieser Übersicht wurde sich auf Anwendungen in den biometrischen Feldern der Iris-, Fingerabdruck- und Gesichterkennung beschränkt. Die Aufstellung hat nicht den Anspruch vollständig zu sein. Es wurde aber darauf geachtet, die Bandbreite der verschiedenen Einsatzmöglichkeiten von Wavelets, soweit diese publiziert wurden, vorzustellen.

3.1 Wavelets in der Iriserkennung

3.1.1 Der Algorithmus von John Daugman

John Daugman stellt in einer Reihe von Artikeln [Daug88, Daug93, Daug99, Daug01] einen Algorithmus zur Iriserkennung vor, dessen Basis die Verwendung von Gaborwavelets (siehe 2.1.8) bildet. Dieser Algorithmus wurde 1994 patentiert [Patent] und findet in den kommerziellen Anwendungen der Firma „Iridian[1]“ Verwendung.

Der Algorithmus kann in folgende Schritte unterteilt werden:

1. Lokalisierung des Mittelpunktes der Iris
2. Feststellung der Radien der Irisränder

[1] siehe http://www.iridiantech.com/

3. Transformation des Irisringes in eine Darstellung bezüglich eines pseudo-polaren Koordinatensystems

4. Berechnung des Iriscodes durch 2D-Gaborwavelets

Wie die folgenden Erläuterungen zeigen werden, entspricht diese Struktur nicht explizit der Darstellung Daugmans.

Vielmehr wurde der Algorithmus für die obige Aufstellung der Arbeitsschritte in logische Einheiten aufgebrochen, um so eine Betrachtung zu ermöglichen, die ein klareres Verständnis fördert. Ein weiteres Ziel dieser Darstellung ist eine bessere Vergleichbarkeit mit anderen biometrischen Verfahren, die ebenfalls Wavelets verwenden.

Einige der oben angegebenen Schritte finden in einem Arbeitsgang statt (wie die Lokalisierung des Mittelpunktes und die Feststellung des Pupillenradius) oder sind in anderen Phasen der Verarbeitung implizit enthalten (z. B. ist die Transformation der Irisdaten in das pseudo-polare Koordinatensystem integrierter Bestandteil der Signaturberechnung).

Vorverarbeitung

Das Finden des Mittelpunktes der Iris und das Ermitteln der Grenze zwischen Pupille und Iris geschieht bei dem Verfahren Daugmans mittels eines integro-differentiellen Operators der folgenden Form [Daug93, S. 1149]:

$$\max_{(r,x_0,y_0)} \left\| \frac{\partial}{\partial r} \oint_{r,x_0,y_0} I(x,y)ds \right\| . \tag{3.1}$$

Da weder der Operator noch seine Aufgabenstellung von hoher Relevanz für diese Arbeit sind, wird die Funktionsweise des Operators lediglich kurz beschrieben.

Die Grundidee des Vorgehens ist es, dass im Farbverlauf des menschlichen Auges zwei Grenzen existieren, deren Übergang jeweils durch einen starken Wechsel in der Helligkeit der Bilddaten charakterisiert ist. Von dem Mittelpunkt der Pupille ausgehend befindet sich die erste Grenze dieser Art an der Stelle des Übergangs von Pupille zur Iris und die zweite an der Stelle des Übergangs von Iris zu Sklera. Bei einer Repräsentation der Bilddaten des Auges, bei der hellere Bildpunkte einem größeren Wert entsprechen (z. B. einer Repräsentation als Graustufenbild, in welcher der Wert 0 mit Schwarz und der Wert 1 mit Weiß korrespondiert), findet in der Grenzregion ein sprunghafter Anstieg der Farbwerte statt. Eine Lokalisierung dieser Grenzen über die Verwendung von Kreisintegralen motiviert sich durch den annähernd kreisförmigen Verlauf der beiden Grenzen.

Zuerst wird die Grenze zwischen der Pupille und der Iris gesucht. Ausgehend von einem approximierten Pupillenmittelpunkt werden sukzessiv größer werdende Kreisintegrale (also Kreisintegrale mit wachsendem Radius) berechnet. Mit größer werdendem Radius wächst der Betrag des Integrales. War die Approximation des Pupillenmittelpunktes hinreichend genau, kommt es bei den Beträgen der errechneten Kreisintegrale zu einem Sprung, wenn der Übergang von Pupillen- zu Irisdaten stattfindet, da der Durchschnittswert der Irisdaten durch ihre größere Helligkeit über dem der Pupillendaten liegt.

Dieser Sprung wird über die in dem Operator (3.1) enthaltenen Ableitung identifiziert, da der Betrag der Ableitung an der Stelle des stärksten Sprungs am größten ist. Die Glättungsfunktion ($G_\sigma(r)$) findet Verwendung, um so Fehler durch Bildartefakte zu minimieren und die Feinheit der Analyse festzulegen.

Das Ergebnis einer Anwendung des Operators (3.1) ist ein Radius r, der bezüglich eines Punktes (x_0, y_0) eine Näherung des gesuchten Übergangs angibt. Dieser Radius wird seinerseits verwendet, um mittels Gradientenabstieg eine bessere Approximation des Pupillenmittelpunktes zu berechnen. Dieser neu angenäherte Mittelpunkt wird seinerseits für eine weitere Anwendung des Operators (3.1) genutzt. Dieses iterative Vorgehen zur Bestimmung der Grenze Pupille/Iris und des Pupillenmittelpunkts bricht ab, wenn die Approximation des Mittelpunktes hinreichend gut erscheint (d. h. wenn sich eine weitere Näherung nicht mehr stark von der aktuellen Näherung unterscheidet). Daugman bezeichnet dieses Vorgehend als die Methode der „wachsenden Kreise“ („exploding circles“) [Patent].

Da die Grenze zwischen der Iris und der Sklera des menschlichen Auges häufig unklarer zu erkennen ist (weil sie teilweise von Unter- und Oberlied verdeckt ist), gibt Daugman in [Patent] ein weiteres Verfahren an, um diese Grenze zu finden. Diese Verfahren ist mit dem oben beschriebenen verwandt, aber nicht identisch. Der interessierte Leser sei für Details auf die zitierte Literatur verwiesen.

Analyse der Formel John Daugmans

Um die 256 Byte große Signatur (von Daugman als „Iriscode“ bezeichnet) zu berechnen, findet folgende Formel Verwendung:

$$h_{\{Re,Im\}} = sgn_{\{Re,Im\}} \int_\rho \int_\phi I(\rho,\phi)\; e^{-i\omega(\theta_0-\phi)} e^{-(\tau_0-\rho)^2/\alpha^2} e^{-(\theta_0-\phi)^2/\beta^2} \rho \; d\rho \; d\phi. \tag{3.2}$$

Jeder Berechnungschritt, der mit dieser Formel ausgeführt wird, setzt zwei Bits des Iriscodes. Für das Setzen der Bits ist der Signum-Operator (sgn) verantwortlich. Dieser nimmt per Definition eine reelle Zahl an und gibt,

je nach Vorzeichen dieser Zahl, entweder 1, falls die Zahl positiv ist, oder 0, falls die Zahl kleiner oder gleich Null ist, zurück. In der oben angegeben Formel wird der Operator auf eine komplexe Zahl angewendet. Dieses entspricht einer jeweils separaten Anwendung des Operators auf den Real- und den Imaginärteil der Zahl. Deswegen ist das Ergebnis einer Anwendung des Operators $sgn_{\{Re,Im\}}$ ein Zwei-Tupel:

$$sgn_{\{Re,Im\}}(x + iy) = [sgn(x), sgn(y)]. \tag{3.3}$$

Bei der Betrachtung der Formel fällt auf, dass die Bilddaten nicht über die übliche kartesische Darstellung $I(x, y)$ vorliegen, sondern über eine Polarkoordinaten-Darstellung[2] $I(\rho, \phi)$, wobei ρ für den Betrag und ϕ für die Phase der komplexen Zahl $x + iy$ steht. Ebenso wie die Bilddaten werden auch die Wavelets über Polarkoordinaten parametrisiert.

Im weiteren finden folgende Parameter in der Formel Verwendung:

- α, β: Skalierungsparameter des zweidimensionalen Gaborwavelets
- θ_0, τ_0: Position des zweidimensionalen Gaborwavelets
- ω: Mittlere Frequenz des zweidimensionalen Gaborwavelets

Transformation der Irisdaten

Auch wenn eine explizite Transformation der Bilddaten nicht stattfindet, sondern die angegebenen Formeln direkt mit Pseudo-Polarkoordinaten arbeiten, erleichtert eine vorgezogene Transformation der Bilddaten das Verständnis des Algorithmus. Mathematisch gesehen existiert kein Unterschied, ob die Transformation explizit oder implizit vorgenommen wird[3].

Es findet eine Abbildung des Irisringes auf eine beschränkte Ebene statt. Der Wertebereich der Achsen der Ebene ist wie folgt eingeschränkt:

$$(\rho, \phi) \in [0, 1] \times [0, 2\pi) \subset \mathbb{R} \times \mathbb{R}$$

[2] Hierbei handelt es sich nicht um die tatsächlichen Polarkoordinaten der Bildpunkte. Näheres zur Interpretation der Schreibweise wird in den folgenden Abschnitten erläutert.

[3] Es muss erwähnt werden, dass die mathematische Äquivalenz nur im Fall absoluter Genauigkeit gegeben ist. Bei der Implementation in Computersystemen, mit der im Allgemeinen eine Diskretisierung der Berechnung einhergeht, können durch die mit dieser Diskretisierung entstehenden Ungenauigkeiten unterschiedliche Ergebnisse am Ende der Berechnung stehen, je nachdem, ob implizit (also durch eine direkte Ansprache der Bildpunkte über Polarkoordinaten) oder explizit (also über eine vorhergehende Wandlung des Bildformats) transformiert wurde.

Die Beschränkung der Werte der Phase ϕ liegt in der Natur der Polarkoordinaten. Die Beschränkung des Betrags ρ erfolgt auf der Basis, dass, unabhängig von der tatsächlichen Größe der Iris und ihrer Pupillendilatation, Punkten, die direkt an der Grenze Pupille/Iris liegen, eine Koordinate der Form $(0, \cdot)$ zugeordnet wird. Punkte, die an der Grenze Iris/Sklera liegen, erhalten Koordinaten der Form $(1, \cdot)$. Punkte, die in dem Bereich zwischen den beiden Grenzen liegen, erhalten in ihrer ersten Koordinate Werte aus dem Intervall $[0, 1]$.

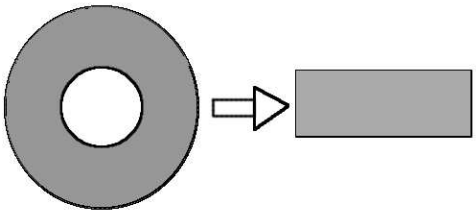


Abbildung 3.1: Transformation des Irisrings

Die Abbildung des Irisbildes $I(x, y)$ in die Darstellung bezüglich (ρ, ϕ) kann wie folgt angegeben werden

$$I(x(\rho, \phi), y(\rho, \phi)) \rightarrow I(\rho, \phi), \tag{3.4}$$

wobei $x(\rho, \phi)$ und $y(\rho, \phi)$ Linearkombinationen aus den Punkten der Grenzen des Irisringes darstellen. Dabei entsprechen Punkte der Form $(x_p(\phi), y_p(\phi))$ Punkten, die auf der Grenze Pupille/Iris liegen. Analog werden die Punkte der Grenze Iris/Sklera mit $(x_s(\phi), y_s(\phi))$ angegeben. Der Winkel ϕ bezieht sich dabei stets auf den entsprechenden Mittelpunkt, also im Falle der (x_p, y_p) auf den der Pupille und im Falle von (x_s, y_s) auf den der Iris [Daug93, S. 1152]. Somit gilt

$$x(\rho, \phi) = (1 - \rho)x_p(\phi) + \rho x_s(\phi) \tag{3.5}$$

und

$$y(\rho, \phi) = (1 - \rho)y_p(\phi) + \rho y_s(\phi). \tag{3.6}$$

Beispiel 3.1 Ein Punkt mit den Koordinaten $(0.5, \phi)$ liegt somit genau in der Mitte der Strecke zwischen den Punkten $x_p(\phi)$ und $x_s(\phi)$, die ihrerseits jeweils auf einem der beiden Ränder des Irisrings liegen und einen Winkel von ϕ zum jeweiligen Mittelpunkte haben.

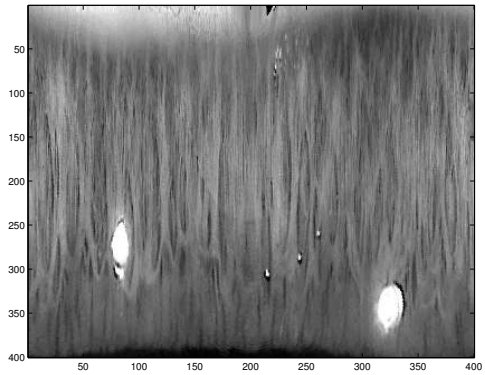


Abbildung 3.2: Ein transformierter Irisring

Daugman spricht in Bezug auf diese Abbildung bzw. Transformation von einem „doubley-dimensionless“ pseudo-polaren Koordinaten-System [Daug99, S. 107].

Bemerkung 3.1 (Pseudo-polares Koordinatensystem)
Der Begriff der „pseudo-polaren Koordinaten“ erklärt sich aus der Transformationformel (3.4), die gegebenenfalls eine nicht-konzentrische Transformation vornimmt. Der radiale Abstand zweier Punkte $(1, \phi_1)$ und $(1, \phi_2)$ zum Ursprung kann somit unterschiedlich sein. Einen weiteren Unterschied zu einer Darstellung in Polarkoordinaten, die man durch eine Verschiebung des Ursprungs des Koordinatensystems auf den Irismittelpunkt erhalten würde, bilden die Punkte mit Koordinaten der Form $(0, \phi)$. Diese Punkte würden in einer Darstellung mit reinen Polarkoordinaten mit dem Punkt $(0, 0)$ identifiziert werden, da sie einen radialen Abstand von 0 zum Ursprung haben. In der gewählten pseudo-polaren Schreibweise (ρ, ϕ) stellt die Koordinate ρ lediglich einen Offset zu der Grenze zwischen Iris und Pupille dar.

Die Verwendung dieser Pseudo-Polarkoordinaten zur Identifizierung von Bildpunkten $(I(\rho, \phi))$ weist gegenüber dem kartesischen Koordinatensystem für die Anwendung in der Iriserkennung folgende Vorteile auf (s.u.).

Eines der Hauptprobleme des Vergleichs digitaler Bilddaten ist die zuverlässige Zuordnung korrespondierender Bildpunkte zweier Abbildungen desselben Objektes. Um einen Vergleich der Irismuster zweier Augen vornehmen zu können, muss gewährleistet sein, dass im Laufe des Vergleichs stets die gleichen Regionen der Iris miteinander verglichen werden. Diese wird durch die Definition des ringförmigen pseudo-polaren Koordinatensystems ermöglicht.

Durch die Verwendung der gewählten Darstellung wird ein ringförmiges Koordinatensystem auf dem Irisring definiert, welches für die Anwendung der

Iriserkennug optimiert ist. Dabei stehen besonders folgende Eigenschaften im Vordergrund:

- Unabhängigkeit von der Pupillendilatation:

  Je nach den Beleuchtungsverhältnissen, unter denen das Bild der Iris aufgenommen wurde, variiert das Größenverhältnis von Pupille und Iris. Bei Lichtverhältnissen von geringer Beleuchtung erweitert sich die Pupille des menschlichen Auges, bei hoher Helligkeit des Umfeldes zieht sie sich zusammen. Somit variiert die Breite des Irisringes.

  Die Transformation in das pseudo-polare Koordinatensystem garantiert, unabhängig von der tatsächlichen Breite des Irisringes, eine robuste Projektion des Koordinatensystems auf den Irisring.

- Implizite Normalisierung der Irisdaten („size invariance“):

  Je nach Abstand des aufgenommenen Auges zur Kamera variiert die Größe der Iris im untersuchten Bild. Ebenso können unterschiedliche Aufnahmeverfahren zu unterschiedlichen Auflösungen der Bilddaten führen. Um diese Einflüsse zu kompensieren sind somit Maßnahmen vor der Analyse der Bilddaten nötig.

  Durch die Transformation auf den festen Bereich $[0, 1] \times [0, 2\pi)$ wird eine explizite Normalisierung der Bilddaten überflüssig. Losgelöst von der tatsächlichen Größe der Iris bzw. des Abbildes der Iris ist eine Zuordnung der Bildpunkte des Abbildes zu Koordinaten robust für verschiedene Abbilder einer Iris.

- Anpassung der Analyse an die radiale Struktur der Bilddaten:

  Ein Großteil der Irismuster (wie z. B. die des Iriskranzes) hat eine radiale Ausrichtung. Das ringförmige Koordinatensystem unterstützt eine Untersuchung derartiger Strukturen.

Berechnung des Iriscodes

Anhand des kreisförmigen Koordinatensystems werden eine Anzahl von so genannten *Analyse-Bändern* („analysis bands“ [Patent]) definiert. Diese Bänder haben jeweils einen festen Radius ρ bezüglich der pseudo-polaren Darstellung. Anhand dieser Bänder positionieren sich die Punkte, die mit verschieden parametrisierten zweidimensionalen Gabor-Wavelets (siehe Formel (3.1.1)) untersucht werden. Der untersuchte Punkt bildet hierbei jeweils den Mittelpunkt des zweidimensionalen Wavelets. Wie oben bereits beschrieben, basiert die Generierung von zwei Bits des Iriscodes auf der Berechnung eines Waveletkoeffizienten.

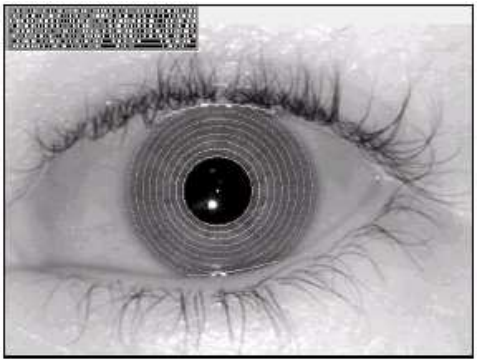


Abbildung 3.3: Visualisierung der Analyse-Bänder (aus [Daug99])

Daugman motiviert die Wahl dieser Codierung wie folgt [Daug99]: Der komplexe Waveletkoeffizient wird als Vektor in der zweidimensionalen Ebene interpretiert. Die beiden Bits repräsentieren dabei, in welchem Quadranten der Ebene der Vektor liegt (siehe Abb. 3.4). Somit bestimmt lediglich die Phase, aber nicht der Betrag der Koeffizienten, den Iriscode.

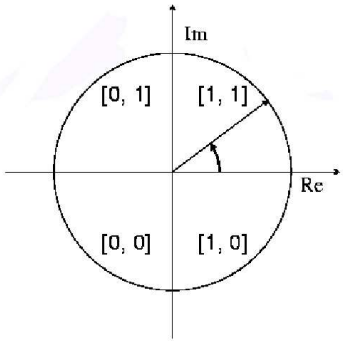


Abbildung 3.4: Abbildung der Waveletkoeffizienten auf den Iriscode [Daug93]

Genauere Untersuchungen zu der Parametrisierung der Gabor-Wavelets folgen in Abschnitt (4.4).

Bemerkung 3.2 (Multiplikation mit ρ)
Bei einer Aufmerksamen Betrachtung der Formel (3.1.1) fällt auf, dass das Produkt aus Bilddaten und Wavelet weiterhin mit den Faktor ρ multipliziert wird. Keine der Veröffentlichungen Daugmans erwähnt diesen Faktor. Der Wertebereich von ρ erstreckt sich auf das Intervall $[0, 1]$. Somit haben bei der

Berechnung eines Waveletkoeffizientens die Bildpunkte die näher am Pupillenmittelpunkt liegen weniger Gewicht, als solche, die weiter außen liegen. Diese Gewichtung bezieht sich wohlgemerkt auf die Berechnung eines einzelnen Koeffizienten. Das Verhältnis der Koeffizienten untereinander (betreffs ihres Gewichtes in der Signatur) bleibt unberührt.

Bemerkung 3.3 (Größe des Iriscodes)
Die Wahl der Größe des Iriscodes mit 256 Byte gründet laut [Daug93, S. 1153] auf der praktischen Überlegung, dass diese Größe ungefähr der Kapazität des Magnetstreifens einer drei-kanaligen Standard-Kreditkarte nach ISO-7811 entspricht.

Vergleich der Signaturen

Zum Vergleich zweier Iriscodes wird der Hamming-Abstand der beiden Codes benutzt. Der Hamming-Abstand ist ein Abstandsmaß zweier Bitfolgen. Mit Hilfe des Hamming-Abstandes kann auf dem Raum der Bitfolgen einer bestimmten Länge eine Metrik (siehe Definition (B.4)) erklärt werden.

Definition 3.1 (Hamming-Abstand) *Der Hamming-Abstand zweier Bitfolgen A, B der Länge n berechnet sich wie folgt:*

$$HD = \frac{\|A \otimes B\|}{n},$$

wobei die Anwendung des Operators $\|\cdot\|$ auf eine Bitfolge als Summe der gesetzten Bits der Folge interpretiert wird.

Beispiel 3.2 (Hamming-Abstand)
Der Hamming-Abstand der Bitfolgen 1010 und 1100 ist somit

$$\mathrm{HD} = \frac{\|1010 \otimes 1100\|}{4} = \frac{\|0110\|}{4} = \frac{2}{4} = 0.5. \tag{3.7}$$

Der Hamming-Abstand zweier gleicher Bitfolgen ist gleich 0. Der Hamming-Abstand von zwei Bitfolgen, die sich in jedem Bit unterscheiden, ist gleich 1. Der Hamming-Abstand zweier rein zufälliger Bitfolgen ist im statistischen Mittel 0.5.

Um die Einflüsse von störenden Bildelementen wie Augenlidern, Wimpern und Lichtreflexionen auszuschließen, werden diese im Laufe der Bildverarbeitung lokalisiert. Anhand der Kenntnis der Position eventuell vorhandener Bildstörungen erfolgt eine Generierung von jeweils einer Bitmaske pro untersuchtem Irisbild. Ein Bit dieser Maske korrespondiert mit einer Region der Iris. Nicht gestörte Regionen der Iris werden dabei mit dem Wert 1, gestörte mit dem Wert 0 versehen. Die Masken finden wie folgt Verwendung:

$$\mathrm{HD} = \frac{\|A \otimes B \cap maskA \cap maskB\|}{\|maskA \cap maskB\|}. \tag{3.8}$$

Je kleiner der Hamming-Abstand zweier Iriscodes ist, desto ähnlicher sind sich die beiden Iriden. Somit kann ein Schwellwert c festgelegt werden, so dass ein Hamming-Abstand, der kleiner als der Schwellwert ($\mathrm{HD} < c$) ist, als Indiz für den Vergleich zweier Bilder derselben Iris gewertet wird.

Die Neigung des Kopfes im Bezug zur Kamera kann variieren. Daraus folgt, dass verschiedene Aufnahmen derselben Iris einen unterschiedlichen Grad von Drehung der Irisdaten aufweisen können. Um diesen Effekt auszugleichen, berechnet Daugman nicht nur einen Hamming-Abstand zwischen der aufgenommenen Iris und der Referenzsignatur. Im Rahmen der Signaturberechnung wird nicht nur eine Signatur der betrachteten Iris ermittelt, sondern eine Reihe - jede davon bezüglich einer leicht gedrehten Iris. Die Referenzsignatur wird gegen jede dieser Signaturen geprüft, und zur Schwellwertbetrachtung kommt lediglich der Hamming-Abstand der Irisaufnahme, deren Signatur der Referenzsignatur am ähnlichsten ist.

3.1.2 Iriserkennung nach Zhu et al.

In dem Artikel [Zhu00a] beschreiben die Autoren ein Verfahren zur Iriserkennung. Der beschriebene Algorithmus verwendet einen zweigleisigen Ansatz, der sowohl auf Gaborfilter wie auch auf die Wavelettransformation zurückgreift.

Vorverarbeitung

Die Vorverarbeitung der Irisbilder gleicht der Methode J. Daugmans, die in Abschnitt 3.1.1 detailliert beschrieben wurde.

Im Anschluss an die Lokalisierung und Transformation des Irisrings verwenden die Autoren Methoden, um die Bildqualität zu verbessern und Effekte ungleichmäßiger Beleuchtung auszugleichen.

Verwendung der Gaborfilter

Der Algorithmus verwendet ein Paar von gerichteten Gabor-Filtern:

$$h_e(x, y) = g(x, y) \cos\left[2\pi\xi(x \cos\theta + y \sin\theta)\right] \tag{3.9}$$

$$h_o(x, y) = g(x, y) \sin\left[2\pi\xi(x \cos\theta + y \sin\theta)\right] \tag{3.10}$$

wobei die Symbole jeweils folgende Bedeutung haben:

- $g(x.y)$: zweidimensionale Gauß-Funktion
- θ: Rotationsparameter, $\theta \in \left\{0^0, 45^0, 90^0, 135^0\right\}$
- ξ: mittlere Frequenz, $\xi \in \{2, 4, 8, 16, 32, 64\}$

Für jeweils jede Frequenz ξ und jeden Rotationschritt θ findet eine Filterung der Bilddaten statt. Für jedes der resultierenden 24 gefilterten Bilder wird der Mittelwert und die zugehörige Standardabweichung der Bilddaten berechnet. Die insgesamt 48 Ergebnisse dieser Berechnung bilden den ersten Teil der biometrischen Signatur.

Zweidimensionale Wavelettransformation

Als zweiter Analyseschritt findet eine zweidimensionale Wavelettransformation der Bilddaten statt. Anstelle der Verwendung von zweidimensionalen Wavelets benutzt der Algorithmus zwei separate eindimensionale Wavelettransformationen. Die für die Wavelettransformation verwendete Waveletfamilie ist eine Familie, die eine orthogonale Basis bildet (DAUB4). Dieses legt den Schluss nahe, dass eine diskrete Wavelettransformation vorgenommen wird[4]. Von den resultierenden „Sub-Images“ der verschiedenen Auflösungsstufen wird jeweils der Mittelwert und die zugehörige Standardabweichung (siehe Anhang B.3) berechnet. Insgesamt 26 dieser Werte bilden den zweiten Teil der biometrischen Signatur.

Einlernen einer Iris

Zum Einlernen einer Iris k in das System werden eine Reihe von unterschiedlichen Bildern dieser Iris verwendet[5]. Für jedes dieser Bilder wird die Signatur berechnet. Im Anschluss werden diese Signaturen zusammengefasst. Für jedes Element („Feature“) f_i der Signatur (also jedes Einzelergebnis) wird der Mittelwert $f_i^{(k)}$ und die Standardabweichung $\delta_i^{(k)}$ innerhalb der Menge der Signaturen dieser Iris berechnet. Diese Werte kommen beim Vergleich zum Einsatz.

[4] Der den Algorithmus beschreibende Artikel [Zhu00a] spart in seinen Ausführungen viele Details aus, so dass eine genauere Analyse des Vorgehens nicht möglich ist.

[5] Die Autoren verwendeten für ihre Tests des Algorithmus fünf verschiedene Bilder pro Iris.

Vergleich der Signaturen

Zum Vergleich der Signatur einer unbekannten Iris mit den eingelernten Signaturen wird ein gewichteter Euklidischer Abstand („Weighted Euclidean Distance“) genutzt:

$$\mathrm{WED}(k) = \sum_{i=1}^{N} \frac{(f_i - f_i^{(k)})^2}{(\delta_i^{(k)})^2}. \tag{3.11}$$

Hierbei steht f_i für das $i$te Element der Signatur der unbekannten Iris und N für die Länge der Signatur. Der Index k identifiziert eine eingelernte Iris. Je kleiner $\mathrm{WED}(k)$, desto ähnlicher ist die untersuchte Iris der Iris k. Somit kann ein Schwellwert definiert werden, für den zwei Signaturen als zu der selben Iris gehörig erkannt werden können.

3.1.3 Weitere Anwendungen im Bereich der Iriserkennung

Iriserkennung nach Lim et al.

Das in [Lim01] vorgestellte Verfahren verwendet für die Analyse der Irisdaten eine vierstufige diskrete Wavelettransformation. Die Transformation wird mit dem Haarwavelet (siehe Beispiel 2.5) durchgeführt. Es werden keine zweidimensionalen Wavelets benutzt, sondern die Analyse vollführt sequentiell zwei eindimensionale Transformationen. Die errechneten Waveletkoeffizienten werden im Anschluss mit einen Verfahren, das dem entsprechenden Vorgehen von Daugman ähnelt, auf ein binäres Format gebracht. Für den Vergleich der resultierenden Signaturen werden Methoden aus dem Bereich der neuronalen Netze angewendet.

Iriserkennung nach Boles et al.

Der im Artikel [Bole98] vorgestellte Ansatz zur Iriserkennung verwendet ebenfalls Methoden der diskreten, eindimensionalen Wavelettransformation. Zum Vergleich der Waveletkoeffizienten werden Methoden aus dem Gebiet der Statistik verwendet.

3.2 Wavelets in der Fingerabdruckerkennung

3.2.1 Fingerabdruckanalyse von Y. Hamamoto

In seinem Artikel [Hama99] stellt Y. Hamamoto ein auf zweidimensionalen Gaborwavelets basierendes Verfahren zur Fingerabdruckerkennung vor. Das Verfahren ähnelt dem Verfahren von John Daugman (Abschnitt 3.1.1).

Ausgehend von einem sogenannten „Core Point“ werden an festgelegten, relativ zu diesem Punkt positionierten Stellen Gabor-Waveletkoeffizienten berechnet, welche die Signatur des Fingerabdrucks bilden. Im Gegensatz zu Daugman arbeitet Hamamoto mit gerichteten Gaborwavelets (siehe Formel 3.12). Hamamoto beschreibt den Vorteil seiner Methode gegenüber den klassischen, auf Erkennung der Muster des Fingerabdruck basierenden, Methoden wie folgt [Hama99, S. 137]:

> „An advantage of the proposed method is that it does not need true minutiae (ridge, forks and ending), compared with other conventional methods. Thus, the proposed method can be applied to those fingerprints where the image quality is low but still readable“.

Vorverarbeitung: Festlegung der zu untersuchenden Punkte

Die Daten, die den zu untersuchenden Fingerabdruck abbilden, bestehen in diesem Verfahren aus zweidimensionalen Bilddaten, also einem Scan eines Bildes des Fingerabdrucks, welches man z. B. durch die Verwendung eines handelsüblichen Fingerabdruckscanners gewinnt.

Wie in den folgenden Abschnitten beschrieben, basiert die Berechnung der Signatur des Fingerabdrucks auf der Verarbeitung der Bilddaten an festgelegten Punkten, den „Sampling Points“. Hamamotos Verfahren verwendet neun Punkte zur Charakterisierung des Fingerabdrucks. Die ausgewählten Punkte lokalisieren die Regionen des Fingerabdruckbildes, die mittels der Wavelettransformation in ein Datenformat gebracht werden, das einen Vergleich erlaubt.

Ähnlich dem Verfahren Daugmans (siehe 3.1.1) ist eine exakte Positionierung der „Sampling Points“ von großer Wichtigkeit, da nur diese ermöglicht, korrespondierende Regionen unterschiedlicher Bilder zu vergleichen.

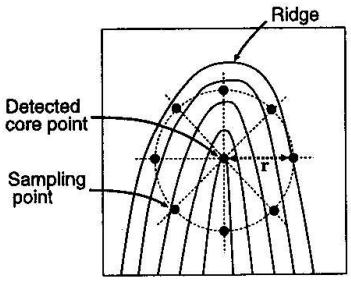


Abbildung 3.5: Ein Fingerabdruck mit eingezeichnetem „Core Point“ und „Sampling Points“ [Hama99]

Die Position der zu verarbeitenden Punkte bestimmt Hamamoto relativ zu einem zentral in die Struktur des Figerabdruckmusters eingebunden Punkt, dem „Core Point“. Dieser Punkt bildet das Zentrum der Wirbelstruktur des Fingerabdrucks. Hamamoto beschreibt keinen eigenen Algorithmus zur Bestimmung dieses Punktes, sondern verwendet einen Algorithmus von A. Shimizu[6]. Die weiteren zu untersuchenden Punkte sind äquidistant auf einem Kreis um den „Core Point“ verteilt (siehe Abb. 3.5). Dieser Kreis hat den Radius r, wobei Hamamoto keine Aussage macht, wie r berechnen ist[7].

Anwendung der Gaborwavelets

Für seine Berechnungen verwendet Hamamoto gerichtete zweidimensionale Gaborwavelets. Das sind Gaborwavelets, die um einem Rotationsfaktor ϑ_k in der xy-Ebene gedreht sind :

$$\psi(x, y, \vartheta_k, \xi) = e^{-\frac{1}{2}\left(\frac{(x\cos\vartheta_k + y\sin\vartheta_k)^2}{\sigma_x^2} + \frac{(-x\sin\vartheta_k + y\cos\vartheta_k)^2}{\sigma_y^2}\right)} \cdot e^{\frac{2\pi(x\cos\vartheta_k + y\sin\vartheta_k)}{\xi} i}. \tag{3.12}$$

Dabei bedeuten die Symbole

- ξ : mittlere Frequenz („wavelength“) des Gaborwavelets
- σ_x, σ_y : räumliche Ausdehnung des zweidimensionalen Gaussfensters. In der Anwendung von Hamamoto werden die Werte von σ_x und σ_y mittels einer Funktion von ξ bestimmt, hängen also von ξ ab.
- ϑ_k : Rotationsparameter, der wie folgt belegt wird:

  $$\vartheta_k = \frac{\pi}{n}(k-1) \qquad , k = 1, 2, \ldots, n$$

  wobei n die Anzahl der Rotationsschritte angibt.

Die Drehung des Gabor-Wavelets erfolgt über die Rotationsmartix des $\mathbb{R}^2$ [Kuip68, S. 58]:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\vartheta_k & \sin\vartheta_k \\ -\sin\vartheta_k & \cos\vartheta_k \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \tag{3.13}$$

[6] Der interessierte Leser sei zur tieferen Einsicht auf den von Hamamoto zitierten Artikel „Detection Method of the Core Point in Fingerprint“von A. Shimizu und M. Hase in *IECE Transactions of the Institute of Electronics and Communication Engineers of Japan, Vol. J67-D(3), pp. 383-384, 1984* verwiesen.

[7] Das Auslassen einer von den Bilddaten abhängigen Berechnung des Radius r legt die Vermutung nahe, dass Hamamoto bereits von einem normalisierten Bildformat ausgeht, also davon, dass die Größe des Abbildes des Fingerabdrucks in einem definierten Verhältnis zu der tatsächlichen Größe des Fingers steht.

Substituiert man in (3.12) x, y durch x', y', um die Rotation aus der Formelschreibweise zu entfernen, erhält man:

$$\psi(x', y', \xi) = e^{-\frac{1}{2}\left(\frac{x'^2}{\sigma_x^2}+\frac{y'^2}{\sigma_y^2}\right)} e^{\frac{2\pi x'}{\xi} i}. \tag{3.14}$$

Berechnung der Signatur

Hamamoto erzeugt die Signatur des Fingerabdrucks, in dem er das ausgerichtete Wavelet ψ an den zu untersuchenden Punkten mit den Bilddaten faltet und den Betrag der entstandenen Funktion berechnet. Das Ergebnis bezeichnet er als „Gabor feature“. Wir werden sehen, dass Hamamotos „Gabor feature“ den in Abschnitt 2.1.4 eingeführten Waveletkoeffizienten entspricht.

Für einen „Sampling Point“ (X, Y), eine Orientierung ϑ und eine Frequenz ξ bestimmt Hamamoto das „Gabor feature“ wie folgt [Hama99, S. 139]:

$$g(X, Y, \vartheta, \xi) = \left| \sum_{x=-X}^{N-X-1} \sum_{y=-Y}^{N-Y-1} I(X+x, Y+y)\psi(x, y, \vartheta, \xi) \right|, \tag{3.15}$$

wobei $I(x, y)$ für die Bilddaten eines $N \times N$ großen Graustufenbildes steht. Die Betragsstriche $|\cdot|$ geben den Absolutbetrag einer komplexen Zahl an.

Die Summierung über $[-X, N-X-1] \times [-Y, N-Y-1]$ zusammen mit der Verschiebung der Bildwerte durch $I(X+x, Y+y)$ führt dazu, dass bei der Faltung das Wavelet ψ bezüglich der Bilddaten über dem „Sampling Point“ (X, Y) positioniert wird. Dieses ist äquivalent zu einer Verschiebung des Nullpunktes der Bilddaten auf (X, Y).

Somit ist (3.15) eine diskretisierte Abwandlung der Wavelettransformation

$$\begin{aligned} g(X, Y, \vartheta, \xi) &= \left| \int_{-X}^{N-X-1} \int_{-Y}^{N-Y-1} I(X+x, Y+y)\psi(x, y, \vartheta, \xi)\, dx\, dy \right| \\ &= \left| \int_{0}^{N-1} \int_{0}^{N-1} I(x, y)\psi(x-X, y-Y, \vartheta, \xi)\, dx\, dy \right|. \end{aligned} \tag{3.16}$$

Die Menge der berechneten „Gabor features“ (also der Waveletkoeffizienten) bildet die Signatur des Fingerabdrucks. Für jeden „Sampling Point“ wird für jede untersuchte Frequenz ξ und jede Orientierung ϑ_k ein Koeffizient berechnet. In seinem Artikel untersucht Hamamoto (inklusive der „Core

Points“) neun Punkte und verwendet dabei acht Orientierungen und eine mittlere Frequenz ξ. Somit besteht seine Signatur aus $9 * 8 * 1 = 72$ Werten.

Die Signatur wird als „Gabor Vector“ bezeichnet.

Vergleich der Signaturen

Von der exakten Positionierung des Zentrumspunktes (d. h. des „Core Point“) sind die Positionen der weiteren zu untersuchenden Punkte unmittelbar abhängig. Somit führt ein fehlerhaft festgelegter Zentrumspunkt zu einer ungeeigneten Signaturberechnung. Da die Güte der Bestimmung der Position des Zentrumspunktes stark von der Qualität des Abbildes des Fingerabrucks abhängt, benutzt Hamamoto eine Methode zum Vergleich der Signaturen, die eine Fehlertoleranz gegenüber Ungenauigkeiten in der Positionsbestimmung des Zentrumspunktes bietet.

Anstatt lediglich eine Signatur des Fingerabdrucks zu berechnen, erzeugt das Verfahren eine ganze Reihe von Signaturen. Jede berechnete Signatur $p_{(\Delta u, \Delta v)}$ benutzt einen leicht verschobenen Zentrumspunkt, wobei $(\Delta u, \Delta v)$ die Verschiebung zum ermittelten Zentrumspunkt p angibt. Die Anzahl der zu berechnenden Signaturen wird durch einen Parameter R bestimmt, der den maximalen Abstand der Verschiebung zum ursprünglichen Zentrumspunkt festlegt.

Für eine maximale Verschiebung R, einen Zentrumspunkt p und eine Referenzsignatur q, gegen die verglichen wird, berechnet sich der Abstand der Signaturen wie folgt:

$$S = \max_{\substack{-R \leq \Delta u \leq R \\ -R \leq \Delta v \leq R}} \frac{p^T_{(\Delta u, \Delta v)} q}{\parallel p_{(\Delta u, \Delta v)} \parallel \parallel q \parallel}. \tag{3.17}$$

Man beachte, dass in dieser Formel die Punkte p und q als Vektoren verwendet werden.

Ist das errechnete Abstandsmaß S (wobei S für „similarity“ steht) größer als ein Schwellwert t, werden die beiden verglichenen Fingerabdrücke als gleich angesehen.

Folgerungen

Das Verfahren zur Fingerabdruckerkennung von Hamamoto ähnelt stark dem von John Daugman zur Iriserkennung. Anhand von festgelegten Punkten werden Signaturen aus Waveletkoeffizienten erzeugt. Die korrespondierenden Werte der einzelnen Signaturen werden paarweise verglichen. Aus diesen Vergleichen wird ein Abstand berechnet. Somit entspricht in diesem

Verfahren ein Signaturenvergleich unmittelbar einem Vergleich der Waveletkoeffizienten.

Wie bei Daugmans Verfahren ist die genaue Positionierung der untersuchten Punkte entscheidend für die Funktionsfähigkeit des Verfahrens. Bei einer ungenauen Positionierung der Punkte kann es dazu kommen, dass „Gabor features“ von verschiedenen Regionen des Fingerabdrucks verglichen werden, wodurch das Verfahren seiner Wirkungsgrundlage beraubt wird.

Ebenfalls vergleichbar zu Daugmans Verfahren findet keine „intelligente“ Analyse der Eigenschaften des Fingerabdrucks statt, d. h. es findet keine semantische Untersuchung der Daten statt. Tatsächlich vorkommende Muster in Fingerabdrücken (wie etwa Minutien) werden nicht als solche identifiziert. Für den Vergleich der Signaturen werden allein die berechneten Waveletkoeffizienten ohne weitere Interpretation benutzt.

3.2.2 Weitere Anwendungen im Bereich der Fingerabdruckerkennung

Kombinierte Fingerabdruck Vorverarbeitung und Erkennung

Im Artikel „Feature selective filtering for ridge extraction“ (siehe [Erol99]) beschreiben die Autoren ein auf Gaborfiltern basierendes Verfahren zur Kantenverstärkung und gleichzeitiger Merkmalserkennung bei Fingerabdrücken. Die Waveleteigenschaften der Gaborfilter spielen in der Beschreibung dieser Anwendung keine Rolle. Es werden gerichtete Gaborfilter verwendet, vergleichbar mit denen des Verfahrens von Hamamoto (Abschnitt 3.2.1).

Orientierungs-sensitive Vorverarbeitung

Die Autoren des Artikels [Pank00] nutzen die Eigenschaft der Orientierungssensitivität der Gaborwavelets aus. Im Prozess der Vorverarbeitung werden mit Hilfe gerichteter Gaborfilter eine Reihe von Varianten des Fingerabdrucks gewonnen, bei denen Bildinformationen der jeweiligen Orientierung (in Korrespondenz zu dem verwendeten Gaborwavelet) betont sind. Diese Methode der Vorverarbeitung unterstützt den Prozess der Minutien-Findung, da die Minutien ihrerseits ebenfalls mit Richtungsinformationen behaftet sind. Durch dieses Vorgehen können Einflüsse von Störungen, hervorgerufen durch mangelhafte Bildqualität, abgemildert werden.

3.3 Wavelets in der Gesichtserkennung

3.3.1 Lokalisierung von charakteristischen Punkten

Der Artikel [Wisk99] beschreibt eine Methode zur Gesichtserkennung mittels elastischer Graphen. Während die eigentliche Vergleichsmethode keine Wavelets nutzt, werden aber Wavelets zur Lokalisierung der charakteristischen Punkte des Bildes verwendet. Diese Punkte, welche die Positionen von Mund, Nase, etc. symbolisieren, bilden die Knoten des Graphen.

Wavelets zur Vorverarbeitung

Wie oben bereits angedeutet, werden in dem beschriebenen Algorithmus die Gaborwavelets zur Vorverarbeitung verwendet: Während bei den Verfahren von Daugman (Abschnitt 3.1.1) oder Hamamoto (Abschnitt 3.2.1), bei denen die Untersuchung der Struktur des betreffenden Bildes durch Methoden, die keine Wavelets verwenden, realisiert wird und erst im Anschluss, nach der Untersuchung der Bildstruktur, zum Vergleich der Bilddaten Wavelets zum Einsatz kommen, verwendet diese Methode Wavelets zur Feststellung der Struktur des Bildes.

Korrespondierende Punkte

Die auf dem Vergleich von elastischen Graphen basierende Methode zur Gesichtserkennung setzt als ersten Arbeitsschritt voraus, dass bei den Bildern der zu vergleichenden Gesichter miteinander korrespondierende Punkte (Mund, Nase, Ohren, Augen, ...) lokalisiert werden. Aus diesen Punkten werden im zweiten Schritt Graphen gebildet, deren Vergleich die Ähnlichkeit zwischen den Gesichtern bestimmt.

Zum Lokalisieren dieser korrespondierenden Punkte setzt das folgende Verfahren Gaborwavelets ein.

Jets

Zur Untersuchung der Bildstruktur führen die Autoren so genannte *Jets* ein. Die Autoren definieren einen Jet als eine Beschreibung der Grauwerte einer kleinen Region des Bildes in der Nachbarschaft eines Bildpunktes (X, Y). Die Jets basieren auf der Wavelettransformation

$$\mathcal{J}_j(X, Y) = \int\int I(x, y)\psi_j(x - X, y - Y)\, dx\, dy \tag{3.18}$$

mit einer Familie von zweidimensionalen Gaborwavelets

$$\psi_j(x,y) = \frac{\xi_k^2}{\sigma^2}\, e^{-\frac{\xi_k^2(x^2+y^2)}{2\sigma^2}} \left[e^{i(\xi_k x \cos\vartheta_l + \xi_k y \sin\vartheta_l)} - e^{-\frac{\sigma^2}{2}} \right] \tag{3.19}$$

mit folgenden Parametern:

- $\xi_k = 2^{-\frac{k+2}{2}}$ mit $k = 0, \ldots, 4$: mittlere Frequenz des Gaborwavelets
- $\vartheta_l = l\frac{\pi}{8}$ mit $l = 0, \ldots, 7$: räumliche Orientierung des Wavelets
- $\sigma = 2\pi$: bestimmt im Zusammenspiel mit ξ die räumliche Ausdehnung des Gauß-Fensters (siehe Abschnitt 2.1.3) und somit die Skalierung des Wavelets

Der Index j berechnet sich aus den Indizes k und l:

$$j = k + 8l$$

Ein Jet $\mathcal{J}$ ist für einen Punkt (X, Y) somit eine Menge $\{\mathcal{J}_j\}$ mit 40 Elementen. Die Elemente von $\mathcal{J}$ sind komplexe Waveletkoeffizienten, welche aus der Wavelettransformation am Punkt (X, Y) resultieren, wobei Versionen des Wavelets (3.19) mit fünf unterschiedlichen Mittenfrequenzen in jeweils acht verschiedenen Orientierungen verwendet wurden.

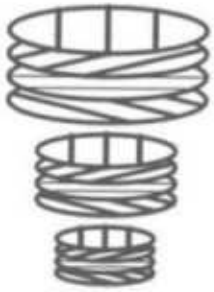


Abbildung 3.6: Schematische Darstellung eines Jets, bestehend aus zwölf Gaborwavelets [Wisk99].

Da die Elemente von $\mathcal{J}$ komplex sind, bietet sich eine Darstellung in Polarkoordinatenschreibweise an, die fortan für die weiteren Betrachtungen verwendet wird:

$$\mathcal{J}_j = a_j\, e^{i\phi}. \tag{3.20}$$

Dabei notiert a_j den Betrag und ϕ die Phase der Koeffizienten.

Identifikation von Bildpunkten mittels Vergleich von Jets

Die Jets werden verwendet, um korrespondierende Bereiche von zu vergleichenden Bildern zu bestimmen. Diesem Verfahren liegt die These zugrunde, dass ähnliche Bilddaten in ähnlichen Jets resultieren.

Die grundsätzliche Idee hinter dem im folgenden beschriebenen Verfahren ist die Verwendung von zwei Jets. Der erste Jet $\mathcal{J}$ dient als Referenz. Er ist in seiner Position zu den Bilddaten festgelegt und beschreibt die Region des Bildes, zu welcher der korrespondierende Bereich zu finden ist (also beschreibt z. B. $\mathcal{J}$ die Nase eines ausgewählten Gesichtes). Der zweite Jet $\mathcal{J}'$ ist in seiner Position zu den Bilddaten variabel. Von einer initialen Position ausgehend, werden seine Ortsparameter sukzessive angepasst, bis die Ähnlichkeit der beiden Jets ein Maximum erreicht hat.

Um die Ähnlichkeit von Jets zu bestimmen, müssen geeignete Funktionen zum Vergleich von Jets definiert werden.

Vergleichmethoden

Die Autoren führen an, dass aufgrund von Phasenverschiebung auch auf identischen Bilddaten nahe beieinander liegende Jets sehr unterschiedliche Koeffizienten haben können, obwohl sie fast den selben Bildbereich beschreiben. Dadurch entstehen beim Vergleich Probleme. Die Autoren stellen zwei Methoden zum Vergleich von Jets vor: Die erste umgeht das eben beschriebene Problem, indem die Phase der Koeffizienten ignoriert wird, die zweite gewinnt aus der Differenz der Phasen weitere Informationen.

Die erste vorgestellte Funktion zum Vergleichen der Jets ignoriert die Phase der Koeffizienten:

$$\mathcal{S}_a(\mathcal{J}, \mathcal{J}') = \frac{\sum\limits_j a_j a'_j}{\sqrt{\sum\limits_j a_j^2 \sum\limits_j a'^2_j}}. \tag{3.21}$$

Für einen festen Jet $\mathcal{J}$ und einen bezüglich der Position des Jets zu den Bilddaten variablen Jet $\mathcal{J}'$ ist die Funktion $\mathcal{S}_a(\mathcal{J}, \mathcal{J}')$ stetig und hat ein Maximum an der Position von $\mathcal{J}$[8]. Somit eignet sich diese Funktion bei räumlich hinreichend nah beieinander liegenden Jets für einen Algorithmus, der sukzessive bessere Näherungen der Position von $\mathcal{J}'$ durch Methoden wie z. B. Gradientenabstieg ermittelt.

Für eine Vergleichsfunktion, welche die Phaseninformationen mit einbezieht, ist es nötig die Verschiebung („displacement“) $d = (d_x, d_y)$ von $\mathcal{J}'$ bezüglich $\mathcal{J}$ zu berücksichigen. Gesetzt den Fall, dass $\mathcal{J}'$ und $\mathcal{J}$ hinreichend nahe

[8] Zumindest falls $\mathcal{J}$ und $\mathcal{J}'$ bezüglich derselben Bilddaten berechnet werden.

beieinander liegen und somit denselben bzw. sich ähnelnde Bildbereiche beschreiben, kann die Phasenverschiebung zwischen den beiden Jets wie folgt ausgeglichen werden [Wisk99, S. 363]:

$$\mathcal{S}_\phi(\mathcal{J},\mathcal{J}') = \frac{\sum\limits_j a_j a'_j \cos(\phi_j - \phi'_j - \xi_k d_x \cos\vartheta_l - \xi_k d_y \sin\vartheta_l)}{\sqrt{\sum\limits_j a_j^2 \sum\limits_j a'^2_j}}. \tag{3.22}$$

Des Weiteren zeigen die Autoren, wie über eine Taylorerweiterung der Formel (3.22) eine Abschätzung von d errechnet werden kann, um so im nächsten Arbeitsschritt des Algorithmus eine bessere Näherung für die Position von $\mathcal{J}'$ finden zu können. Für weitere Details seien interessierte Leser auf den Artikel [Wisk99] verwiesen.

Generierung des Graphen

Zum Vergleich zweier Gesichter werden die spezifischen Eigenschaften eines elastischen Graphens benutzt, dessen Knoten die charakteristischen Punkte eines Gesichtes sind (also Mund, Nase, Augen, ...). Je nach Gesicht und Aufnahme kann jedoch derselbe charakteristische Punkt in unterschiedlichen Jet-Koeffizienten resultieren. So unterscheiden sich z. B. die Jets von geschlossenen Augen signifikant von Jets geöffneter Augen. Aus diesem Grund existieren für jeden zu identifizierenden Punkt eine Reihe von Referenz-Jets, die gemeinsam den Punkt des Gesichts beschreiben, um so die unterschiedlichen Ausprägungen des Merkmals auszugleichen. Die Autoren bezeichnen die Menge der Jets, die demselben Merkmal zugeordnet sind, als „Bunch“. Bei der Analyse der Bilddaten wird das Element eines Jets berücksichtigt, das die größte Ähnlichkeit zu den Bilddaten aufweist. Dieses wird als „Local Expert“ bezeichnet. Der resultierende Graph trägt in Referenz auf die multiplen Merkmalscharakterisierungen den Namen „Face Bunch Graph“ (siehe Abb. 3.7).

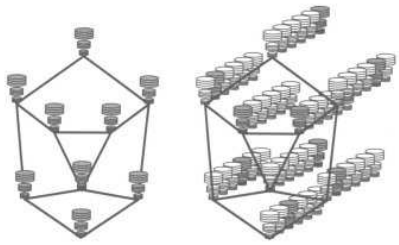


Abbildung 3.7: „Face Bunch Graph“, ohne und mit Markierung des „Local Expert“ [Wisk99].

Zusätzlich zum Vergleich der Eigenschaften der erzeugten Graphen, kommt als zweites Entscheidungskriterium der direkte Vergleich der Jets hinzu. Somit haben die Waveletkoeffizienten einen unmittelbaren Einfluss auf die Signatur des Gesichts.

Folgerungen

Die Grundannahme, auf der das gesamte hier beschriebene Verfahren beruht, lautet, dass die Transformation von vergleichbaren Merkmalen eines Gesichtes (Mund, Nase, Ohren) ähnliche Koeffizienten der resultierenden Jets zur Folge hat. Das bedeutet, dass die Waveletkoeffizienten dieser Merkmale sich gleichen.

Durch die Verwendung der Mengen von Jets für ein Merkmal (den „Bunchs“) steht ein Mittel zur Kalibrierung des Systems zur Verfügung. Der Algorithmus kann der Klasse der Gesichter, die erkannt werden soll, angepasst werden. Diese Eigenschaft könnte sich im praktischen Einsatz jedoch als Nachteil erweisen. Je genereller die Klasse der Gesichter gefasst wird, desto größer ist die Anzahl der Jets in den einzelnen „Bunchs“. Je stärker jedoch die Ausprägung der Jets innerhalb eines „Bunchs“variiert, desto größer ist eventuell die Fehleranfälligkeit des Verfahrens.

Die Auswirkung dieser eventuellen Fehleranfälligkeit kann abgeschätzt werden: Bei einem Wachsen der Fehlerrate des Algorithmus, kann mit einem Wachsen der FRR (siehe Abschnitt 2.3.3), aber nicht mit einem starken Wachsen der FAR gerechnet werden. Mit Abnahme der Genauigkeit der Vorverarbeitung nimmt die Qualität der Graphen ab. Aufgrund einer fehlerhaften Lokalisierung der charakteristischen Punkte findet eine fehlerhafte Zuordnung der Punkte zu Regionen des Gesichtes statt (im Extremfall könnte z. B. ein Auge für ein Ohr gehalten werden)[9]. Dieses führt zu einem entarteten Graphen, dessen Charakteristik sich mit hoher Wahrscheinlichkeit von der eines eingelernten Gesichts unterscheidet. So kann auch die Vorverarbeitung eines eingelernten Gesichts fehlschlagen, was ein negatives Resultat der Erkennung zur Folge hat.

Zur Erzeugung eines „false positive“ andererseits müsste die Fehlpositionierung der Punkte der Struktur eines existierenden Gesichtes gleichen, also z. B. die Position der Augen nur schwach verschoben sein. Unter der Voraussetzung, dass die grundsätzlichen Annahmen bestehen können und Jets ein geeignetes Mittel sind, korrespondierende Punkte in Bilddaten zu identifizieren, erscheint dieses Fehlverhalten vergleichsweise unwahrscheinlich.

Diese aus rein theoretischen Betrachtungen abgeleitete Einschätzung des

[9] Selbst bei stark variierenden Jets innerhalb eines „Bunchs“ist ein derartiger Fehler sehr unwahrscheinlich. Dieses Beispiel soll lediglich in bildlicher Sprache ein Fehlverhalten des Algorithmus beschreiben.

Verhalten des Algorithmus sollte vor einer endgültigen Beurteilung jedoch unbedingt in praktischen Versuchen überprüft werden.

Mit Blick auf die Art der Verwendung der Wavelets in diesem Verfahren kann geschlossen werden, dass auch in diesem Fall zur Analyse der Bilddaten keine Methode verwendet wird, die auf einem abstrakten Verständnis des Bildes beruht. Die korrespondierenden Bildpunkte werden allein anhand der mathematischen Ähnlichkeit ihrer Jets identifiziert.

3.3.2 Gesichtserkennung mit „Wavelet Packets“

In dem Artikel „Wavelet Packet Analysis for Face Recognition“ [Garc00] wird eine Methode zur Gesichtserkennung vorgestellt, die „Wavelet Packets“ verwendet. Der Algorithmus verwendet Waveletkoeffizienten sowohl zur Bestimmung der Bildstruktur als auch zur Signaturerzeugung.

Die Autoren schränken den Wirkungsgrad des Algorithmus auf die Erkennung von Gesichtern, deren Abbilder in einer Position frontal zu Kamera aufgenommen wurden und deren Beleuchtung nicht zu stark variiert, ein.

Anders als in den bisher betrachteten Algorithmen, verwendet die beschriebene Methode keine zweidimensionalen Wavelets. Die Bilddaten werden mittels zweier sequentieller eindimensionaler Wavelettransformationen analysiert, wobei die erste Transformation die Bilddaten in vertikaler Richtung (also entlang der y-Achse) analysiert und die zweite Transformation in horizontaler (also entlang der x-Achse).

Die Autoren verwenden für ihre Analyse „Wavelet Packets“ (siehe Abschnitt 2.1.7). Die Wavelettransformation wird mit den Filtern H und G durchgeführt. Diese leiten sich, wie bei der Multiskalen-Analyse, aus der „Scalingfunction“ und den Wavelets ab. H ist demzufolge ein Tiefpass (abgeleitet aus der „Scalingfunction“) und G ein Bandpass-Filter (abgeleitet aus den Wavelets):

$$A_n = \Big[H_x \star [H_y \star A_{n-1}]_{|2,1}\Big]_{|1,2} \tag{3.23}$$

$$D_{n1} = \Big[H_x \star [G_y \star A_{n-1}]_{|2,1}\Big]_{|1,2} \tag{3.24}$$

$$D_{n2} = \Big[G_x \star [H_y \star A_{n-1}]_{|2,1}\Big]_{|1,2} \tag{3.25}$$

$$D_{n3} = \Big[G_x \star [G_y \star A_{n-1}]_{|2,1}\Big]_{|1,2} \tag{3.26}$$

In diesen Formeln sind die Symbole mit folgenden Bedeutungen belegt:

- A_n: Approximation der Stufe n, wobei $A_0 = I(x, y)$ das Ursprungsbild repräsentiert.

- D_{ni}: Details der Stufe n. Der Parameter i gibt die Ausrichtung der Details an ($i = 1, 2, 3$ steht respektive jeweils für vertikal, horizontal oder diagonal[10]).
- $_{|2,1}$ bzw. $_{|1,2}$: Diese von den Autoren verwendete Notation gibt an, dass sowohl in vertikaler als auch in horizontaler Richtung dyadisch (also in Schritten in Zweierpotenzgröße) transformiert wird.

Der Algorithmus vollführt insgesamt zwei Transformationsschritte. Im ersten Transformationsschritt wird eine Approximation und für jede Richtung (horizontal, vertikal und diagonal) jeweils eine Detaildarstellung berechnet. Da der Algorithmus mit „Wavelet Packets“ arbeitet, findet im zweiten Schritt eine Dekomposition sämtlicher Ergebnisse aus dem ersten Schritt statt, also auch der Detaildarstellungen. Dieses führt zu insgesamt 16 „subimages“: eine Approximation und 15 Detaildarstellungen von verschiedener Auflösung und Richtung (siehe Abb. 3.8).

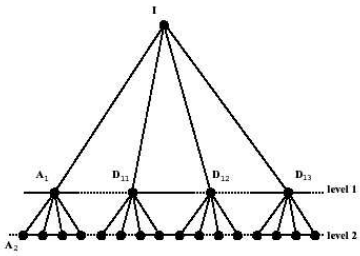


Abbildung 3.8: Zweistufige Analyse mit „Wavelet Packets“ [Garc00]

Identifizierung der Regionen des Gesichts

Die Ergebnisse des ersten Transformationsschritts werden dazu verwendet, das Gesicht in den Bilddaten zu lokalisieren und Regionen des Gesichts zu identifizieren. Dabei wird der Umstand ausgenutzt, dass die Waveletkoeffizienten mit Richtungsinformationen behaftet sind. Die Koeffizienten in D_{11} kodieren Veränderungen des Signals in vertikaler Richtung, D_{12} in horizontaler.

Sukzessive werden zuerst die vertikalen Begrenzungen des Gesichts identifiziert, dann die horizontalen Strukturen (Mundlinie, Nasenbasis, Augenlinie). Die Identifizierung dieser Strukturen funktioniert - vereinfacht ausgedrückt - über die Ermittlung charakteristischer lokaler Betragsmaxima der Waveletkoeffizienten (siehe Abb. 3.9).

[10] Die hier erwähnte Richtung bezieht sich dabei auf die Richtung, in der die Details des Signals berechnet werden.

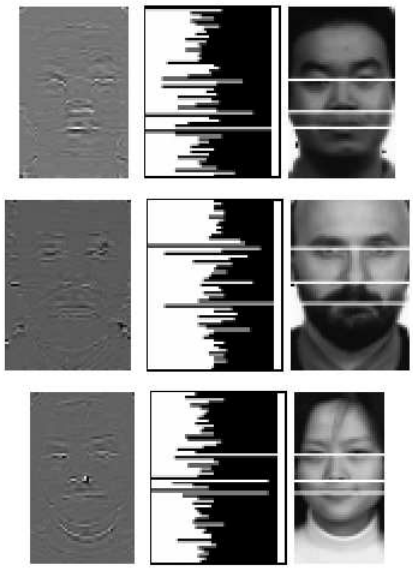


Abbildung 3.9: Identifizierung der Regionen des Gesichts über Koeffizientenmaxima [Garc00]

Über diese Strukturinformationen werden insgesamt vier rechtwinklige Regionen des Gesichtes definiert: die äußeren Regionen des Gesichts („out“), das gesamte Gesicht ohne die äußeren Regionen („in“), die Gesichtsregion oberhalb der Nasenbasis („top“) und die Region unterhalb der Nasenbasis („bottom“) (siehe Abb. 3.10).

Signaturerzeugung

Die Ergebnisse der zweiten Stufe der Transformation (siehe Abb. 3.11) werden verwendet, um die biometrische Signatur zu berechnen. Dieses geschieht mit Hilfe stochastischer Methoden:

Von der Approximation der zweiten Stufe werden für die Regionen „out“, „top“ und „bottom“ jeweils der Mittelwert μ_i und die zugehörige Standardabweichung σ_i der Koeffizienten berechnet. Für die 15 Detailinformationen wird die Standardabweichung für die Vereinigung der Regionen „out“ und „bottom“ ermittelt, was insgesamt zu 21 Werten führt.

Somit wird ein Vektor V gebildet:

$$V = \bigcup_{i=0}^{17} \sigma_i \mu_i, \tag{3.27}$$

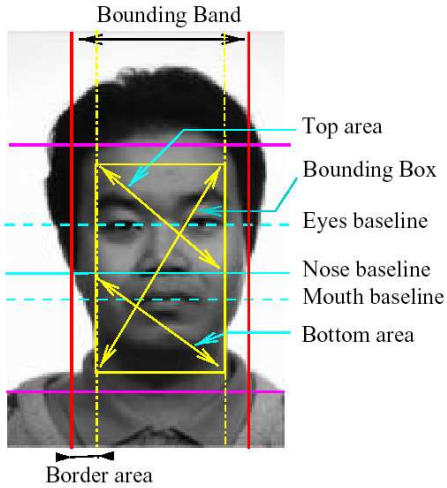


Abbildung 3.10: Die Regionen des Gesichts [Garc00]

wobei für $i \geq 2$ gilt, dass $\mu_i = 0$ ist. Die Indizes $0, 1, 2$ stehen respektive für die Regionen „out“, „top“ und „bottom“ .

Vergleich der Signaturen

Der Vergleich zweier Signaturen geschieht über einen paarweisen Vergleich der Elemente der Vektoren V_k und V_l. Der Abstand des $i$ten Elements der Vektoren berechnet sich wie folgt:

$$D_i(V_k, V_l) = \frac{1}{4}\frac{(\mu_{ik} - \mu_{il})^2}{\sigma_{ik} + \sigma_{il}} + \frac{1}{2}\ln\left[\frac{\frac{1}{2}(\sigma_{ik}^2 + \sigma_{il}^2)}{\sqrt{\sigma_{ik}^2\sigma_{il}^2}}\right], \tag{3.28}$$

wobei σ_{il} für den Wert σ im $i$ten Element des Vektors V_l steht.

Der vollständige Abstand zweier Vektoren und somit zweier Gesichter definiert sich über:

$$D(V_k, V_l) = \sum_{i=1}^{17} D_i(V_k, V_i). \tag{3.29}$$

Je kleiner $D(V_k, V_l)$ ist, desto ähnlicher sind sich die Gesichter.

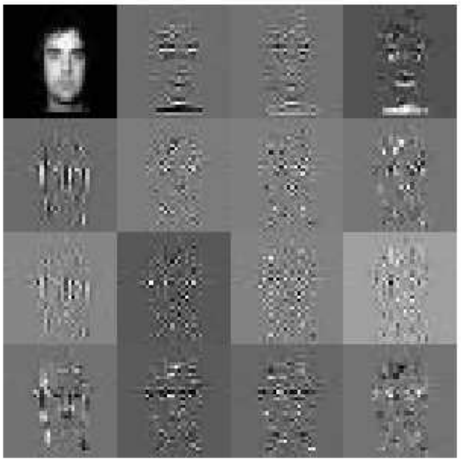


Abbildung 3.11: Darstellung der Ergebnisse des zweiten Transformationsschritts [Garc00]

3.3.3 Weitere Anwendungen im Bereich der Gesichtserkennung

In dem Artikel „Face Recognition Using Independent Gabor Wavelet Features“ [Liu01] stellen die Autoren ein auf Gaborwavelets basierendes Verfahren zur Gesichtserkennung vor. Für die Analyse werden Gaborwavelets mit fünf Skalierungen ν und acht Ausrichtungen μ verwendet. Nach einer erfolgten Faltung des Bildes des Gesichts mit den Wavelets resultiert ein Merkmalsvektor X, bestehend aus den Ergebnissen der 40 Faltungen. Die Faltungsresultate $O_{\nu,\mu}$, also die Ergebnisse der Faltung eines digitalen Bildes mit einem Wavelet, werden für die weiteren Berechnungen als komplexwertige Matrizen interpretiert.

$$X = \left(O^t_{0,0}, O^t_{0,1}, \ldots, O^t_{4,7}\right) \tag{3.30}$$

Auf diesen Vektor X werden fortgeschrittene Methoden der Algebra und Statistik angewandt, um so eine Klassifizierung des Gesichtes durchzuführen, die zum Vergleich bzw. zur Identifikation von Gesichtern verwendet wird.

Kapitel 4

Implementierung und Parametrisierung eines waveletbasierten Iriserkennungsalgorithmus

In diesem Kapitel wird am Beispiel des Algorithmus von John Daugman die Konzeption einer Implementierung und Parametrisierung eines waveletbasierten Algorithmus zur biometrischen Authentikation detailliert beschrieben.

Das Gewicht dieser Darstellung liegt auf dem systematischen Aufdecken und Einschränken der Freiheitsgrade des Algorithmus. Wie später aufgezeigt wird, ist die Anzahl der freien Variablen eines waveletbasierten Algorithmus groß. Die Grundlage einer einsetzbaren Implementierung ist jedoch eine hinreichend gute Parametrisierung dieser Variablen. Ohne ein Verständnis der Bedeutung der Freiheitsgrade des Verfahrens und ihrer Wechselwirkungen untereinander, wird der Versuch einer Parametrisierung schnell zu einem hoffnungslosen Unterfangen. Wie Kapitel 3 entnommen werden kann, enthält keine der Veröffentlichungen der biometrischen Algorithmen ausreichende Informationen über die anzuwendende Belegung der Freiheitsgrade.

Im Anschluss an die Einschränkung der Freiheitsgrade folgen praktische Experimente, die ein iteratives Verfahren verwenden um eine Parametrisierung des Algorithmus zu finden.

Weiterhin werden in diesem Abschnitt Methoden zur Konzeption einer Implementierung vorgestellt, die an die Eigenschaften biometrischer Algorithmen angepasst sind, sowie die angefertigte Implementierung beschrieben.

4.1 Anforderungen an die Implementierung des Algorithmus

Im Vorfeld der Konzeption einer Implementierung sollten stets die speziellen Eigenschaften des Algorithmus bzw. seiner Zielsetzung bestimmt werden. Die Anforderungen des hier beschriebenen Algorithmus leiten sich aus seinem Einsatzgebiet (der biometrischen Authentikation) und seinem mathematischen Fundament (den Wavelets) ab.

4.1.1 Eigenschaften von Algorithmen zur biometrischen Authentikation

Die Anforderungen an die Implementierung biometrischer Algorithmen lassen sich wie folgt gliedern:

Flexibilität

Es kann nicht gewährleistet werden, dass bei der Implementierung eines biometrischen Algorithmus die Umweltbedingungen des späteren Einsatzortes hinreichend genau abgebildet werden. Je nach verwendeter biometrischer Technik kann die Genauigkeit des Algorithmus von verschiedenen Einflüssen der räumlichen Umgebung abhängen.

Somit müssen bei der Konzeption der Implementierung Mittel geschaffen werden, die Kalibrierung der Applikation während der Entwicklung und des Einsatzes anpassen zu können. Genauere Betrachtungen zu diesem Thema folgen in den Abschnitten 4.2.1 und 4.2.2.

Modularisierung

Ein biometrischer Algorithmus gliedert sich in den meisten Fällen in mehrere, algorithmisch unabhängige Stufen. Jede dieser Stufen erfüllt eine spezielle Teilaufgabe, wie z. B. die Bildvorverarbeitung, die Signaturberechnung oder den Signaturvergleich. Eine Implementierung sollte dieser Gliederung Rechnung tragen und eine Modularisierung aufweisen, die dieser angepasst ist. So können die unabhängigen Einzelkomponenten separat getestet, optimiert und den einsatzabhängigen Umständen angepasst werden.

Performanz

Ein wichtiger Aspekt bei der Implementierung von Algorithmen zur Authentikation ist die Performanz. Im Falle der Benutzerauthentikation (z. B. zum Zweck der Zugriffkontrolle) ist die Authentikation ein interaktiver Prozess.

Wenn der Vorgang der Authentikation einen zu langen Zeitraum einnimmt, leiden Benutzerakzeptanz und Handhabbarkeit darunter.

Bemerkung 4.1 (Nicht-interaktive Anwendungen)
Die Anforderung an die Performanz ist anwendungsabhängig. Im Falle eines Einsatzes zur biometrischen Identifikation (siehe Definition 2.18), sind Szenarien denkbar, in denen die Anwendung weder interaktiv noch zeitkritisch ist (z. B. das nachträgliche Untersuchen von Filmmaterial auf das Vorhandensein eingelernter Personen). In diesem Fall kann die oben formulierte Anforderung zugunsten anderer Aspekte (wie z. B. erhöhter Erkennungsgenauigkeit) zurückgestellt werden.

Datensicherheit

Auch wenn der Aspekt der Datensicherheit nicht im algorithmischen Kern enthalten ist, ist er dennnoch von großer Wichtigkeit:

- Im Rahmen der biometrischen Authentikation findet eine Verarbeitung und Speicherung von sensiblen personenbezogenen Daten statt. Ein Schutz dieser Daten ist unerlässlich.
- Weiterhin erfüllen Anwendungen der biometrischen Authentikation häufig Aufgaben der Zugangskontrolle. Somit ist die biometrische Applikation ein Teil der Sicherheitsarchitektur des Systems, in das sie integriert ist (bzw. der Ressource, deren Zugriff sie überwacht).

Aus diesen beiden Punkten folgt die Anforderung an ein klares Konzept zur Absicherung der Übertragung, Verarbeitung und Speicherung der Daten, die im Rahmen der biometrischen Authentikation verwendet und erzeugt werden.

Bemerkung 4.2 (Die vorliegende Implementierung)
Die im Rahmen dieser Arbeit angefertigte Implementierung verwendet keine Methoden, um die soeben formulierte Forderung der Datensicherheit durchzusetzen. Die Implementierung versteht sich vielmehr weitgehend als Prototyp für eine endgültige Entwicklung, bei der dann die genannten Anforderungen berücksichtigt werden sollten.

4.1.2 Eigenschaften waveletbasierter Algorithmen

Algorithmen, deren mathematische Grundlage Wavelets bilden, ist gemein, dass sie eine große Anzahl von variablen Parametern besitzen. Die Qualität einer Implementierung eines Algorithmus hängt vollständig von der Güte der

gewählten Parametrisierung ab. Der Umfang der Freiheitsgrade hängt von dem Einsatzgebiet, der Art der verwendeten Wavelets und dem vorhandenen Vorwissen ab.

Wie in Kapitel 3 dargestellt, enthält ein Großteil der Publikationen von waveletbasierten Algorithmen zur biometrischen Authentikation lediglich die Beschreibung des generellen Verfahrens, aber nur wenige Angaben zur genauen Parametrisierung. Dieses Wissen ist mindestens genauso wertvoll, wie der eigentliche Algorithmus.

Eine Implementierung eines derartigen Algorithmus ist darauf angewiesen, eine schrittweise Verfeinerung der Parametrisierung zu unterstützen. Dieses beinhaltet eine umfangreiche, flexible Schnittstelle und gegebenenfalls einen aussagekräftigen Datenrückkanal zur Evaluation, der über den eigentlichen Einsatzzweck des Algorithmus hinausgeht.

4.2 Unterstützung der Implementierung durch ein Test-Rahmenwerk

In dem Artikel „A Conceptual Framework for Testing Biometric Algorithms within Operating Systems' Authentication“ [Broe02] stellen die Autoren ein umfangreiches, auf die Anforderungen der biometrischen Authentikation zugeschnittenes Rahmenwerk für die Implementierung und den Test von Algorithmen vor. In diesem Rahmenwerk werden sämtliche relevanten Faktoren für eine Implementierung eines Algorithmus zur biometrischen Authentikation erfasst.

Die Konzeption der Implementierung, die in den folgenden Abschnitten vorgestellt wird, orientiert sich an diesem Rahmenwerk.

Bemerkung 4.3 (Einschränkung)
Das Rahmenwerk umfasst sämtliche Phasen einer biometrischen Authentikation und deren Entwicklung und spezifiziert somit auch Aspekte, die nicht in der vorliegenden Aufgabenstellung enthalten sind. Somit kommen nicht alle Teilaspekte des Frameworks mit gleicher Gewichtung zum Tragen. Teilaspekte, die in den folgenden Ausführungen nicht oder nicht vollständig ausgearbeitet werden, wurden im Rahmen der Konzeption bzw. der Implementierung dennoch berücksichtigt, um so eine spätere Weiterentwicklung bzw. Integration der Implementierung zu ermöglichen. Dieses geschah sowohl explizit (durch Hinweise in der Spezifikation oder dem Quelltext) oder implizit (im Rahmen der Konzeption/Modularisierung).

4.2.1 Das Rahmenwerk

Grundlage der folgenden Betrachtungen ist die Erkenntnis, dass es bei der Entwicklung und dem Test eines Algorithmus zur biometrischen Authentikation nicht ausreicht, diese unter Laborbedingungen vorzunehmen. Auch ein Algorithmus, der aufgrund von Untersuchungen in vorhandenen Testumgebungen (z. B. im Entwicklungslabor) bzw. mit Testdatensätzen (wie z. B. einer Bilderdatenbank) als hinreichend gut kalibriert erscheint, kann sich beim Einsatz unter realen Bedingungen als unzureichend herausstellen. Dieses liegt an Aspekten[1], wie

- uneinheitlichen Beleuchtungsbedingungen (z. B. abhängig von den Raumbeschaffenheiten, der Tages- und Jahreszeit, etc.),
- wechselhafter Bildqualität (z. B. durch unterschiedliche Positionierung der Kamera bzw. variierenden Abstand des Merkmals zum Objektiv),
- Fehlerquellen hervorgerufen durch die menschliche Interaktion (z. B. Schwierigkeiten in der Benutzung des Systems, mangelnde Akzeptanz).

Derartige Aspekte sind unter Laborbedingungen nur schwer zu simulieren. Somit müssen bei der Entwicklung eines Algorithmus und dessen Implementierung Mittel geschaffen werden, die Einflüsse der oben aufgeführten Umstände festzustellen und auszugleichen. Das Rahmenwerk stellt für diesen Zweck zwei Methoden zur Verfügung:

- Umfangreiche Datensammlung

  Es wird angestrebt, alle anfallenden Daten aufzuzeichnen, die für die Evaluation bzw. die Verbesserung des Algorithmus relevant sind. Es ist vorgesehen, sowohl quantitative Daten (wie Zeitstempel, Zwischenergebnisse der Berechnungen, etc.) wie auch qualitative Daten (z. B. Beobachtungen über die Nutzerinteraktion) aufzuzeichnen. Dafür sind interne Schnittstellen direkt zum Algorithmus wie auch externe Eingabemöglichkeiten für menschliche Benutzer vorgesehen.

- Iterativer Prozess zur Parametrisierung

  Mit Hilfe der gesammelten Daten findet ein iterativer Vorgang zur Verbesserung der Parametrisierung statt. Dieser ist eingebettet in die vorrausgehende grundsätzliche Festlegung der unveränderlichen Randbedingungen (Einsatzgebiet, biometrische Technik, Zielsystem) und der abschließenden Evaluation (siehe Abbildung 4.1).

[1] Die folgenden Aspekte beziehen sich auf den hier behandelten Fall der Iriserkennung. Andere biometrische Anwendungen können weitere Störungsquellen aufweisen (z. B. wechselhafter Lärmpegel des Umfeldes im Falle der Stimmerkennung).

4.2.2 Ablauf der Entwicklung

Der Ablauf der Entwicklung einer biometrischen Applikation gliedert sich in vier Phasen, die in den folgenden Abschnitten beschrieben werden.

1. Grundlegende Entscheidungen

Bevor mit der eigentlichen Entwicklungsarbeit angefangen werden kann, ist es notwendig die Randbedingungen festzulegen. Diese werden durch Faktoren wie den Einsatzzweck und den Einsatzort der Applikation bestimmt. Festzulegen sind unter anderem die verwendete biometrische Technik (z. B. Iriserekennung), das verwendete Betriebssystem und die ortsabhängigen Eigenschaften (wie die zu erwartenden Beleuchtungsverhältnisse).

2. Implementierung

Im Rahmen der Implementierung ist vorab festzulegen, welcher Algorithmus zum Einsatz kommt. Abhängig von dieser Entscheidung findet eine Modularisierung des Algorithmus statt, die eine Spezifizierung der Schnittstellen und der Module sowie eine Berücksichtigung eines Datenrückkanals beinhaltet. Die Grundlage für die Modularisierung des Algorithmus bildet der in Abschnitt 4.2.3 beschriebene Vorgang der biometrischen Authentikation.

3. Iterative Verfeinerung

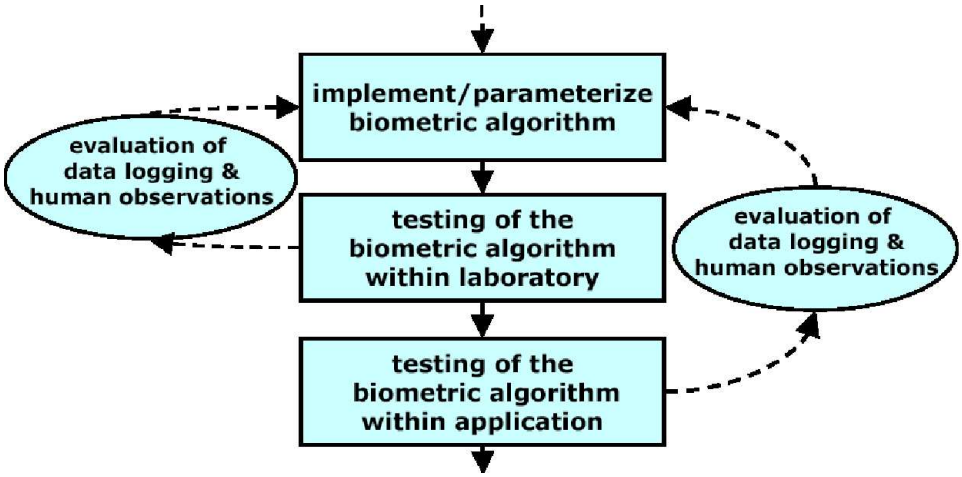


Abbildung 4.1: Iterative Verfeinerung des Algorithmus‘ [Broe02b]

Nach einer Implementierung und vorläufigen Parametrisierung findet ein zweistufiges Vorgehen zur schrittweisen Verbesserung statt. Im ersten Schritt

wird die Implementation/Parametrisierung unter Laborbedingungen getestet und gegebenenfalls angepasst. Die Anpassungen geschehen auf Grundlage der gesammelten Datenbasis. Wenn ein ausreichender Gütegrad erlangt wurde, finden im zweiten Schritt Tests unter Realbedingungen statt. Diese Tests simulieren möglichst genau die Umgebung, die bei einem späteren Einsatz der Implementierung auftreten. Die Tests sollten direkt an einem der zukünftigen Einsatzorte vorgenommen werden und mit Probanden, die aus dem Kreis der potentiellen Benutzer stammen. Neben den Ergebnissen der automatischen Datensammlung kommen hier gleichberechtigt Daten hinzu, die von menschlichen Beobachtern gesammelt wurden, und die z. B. Aspekte wie Mensch-/Maschine-Interaktion oder Benutzerakzeptanz aufzeichnen. Aufgrund der gesammelten Daten werden Parametrisierung und Implementierung gegebenenfalls angepasst und erneut mittels des zweistufigen Verfahrens getestet. Der Vorgang der iterativen Verfeinerung bricht ab, wenn entweder die Güte des Verfahrens den Anforderungen entspricht oder keine Verbesserung mehr erzielt werden kann.

4. End-Evaluation

Am Ende einer Entwicklung steht die End-Evaluation. Hier findet eine endgültige Entscheidung statt, ob die entwickelte Anwendung zur biometrischen Authentikation geeignet ist, unter den definierten Einsatzbedingungen verwendet zu werden. Ausschlaggebend sind neben Gründen algorithmischer Natur, wie dem Erreichen einer ausreichenden Erkennungsrate, auch Aspekte die sich aus dem Zusammenspiel zwischen der Anwendung und den Nutzern ergeben. Auch biometrische Anwendungen zur Authentikation, die sich durch eine zuverlässig hervorragende Erkennungsgenauigkeit auszeichnen, können im täglichen Einsatz nutzlos sein, wenn sie entweder zu umständlich zu bedienen sind, oder bei den menschlichen Benutzern keine Akzeptanz finden[2]. Bei einem negativen Ergebnis der End-Evaluation ist es notwendig zu Phase 1. zurückzukehren, um so eine Anpassung der grundlegenden Randbedingungen vorzunehmen. Dieses kann eine Wahl einer anderen biometrischen Technik oder eine Überarbeitung der Einsatzbedingungen beinhalten. Bei einem positiven Endergebnis der Evaluation kann mit der Entwicklung der endgültigen Anwendung begonnen werden. Diese unterscheidet sich nicht wesentlich von der getesteten Variante, benötigt aber nicht mehr den vollen Umfang der Datensammlung und stellt erweiterte Anforderungen im Bereich der Datensicherheit und Ausführungsgeschwindigkeit.

[2] Ein naheliegendes Beispiel ist in diesem Zusammenhang eine Methode der biometrischen Authentikation, die auf der DNA-Analyse basiert und die Abgabe einiger Tropfen Blut verlangt. Auch wenn diese Methode über ausgezeichnete Erkennungsgenauigkeit und gute Performance verfügen mag, ist anzunehmen, dass ein Einsatz auf täglicher Basis wie z. B. zur Zugangsbeschränkung eines Gebäudes an dem Widerstand der Nutzer scheitern würde.

4.2.3 Der Vorgang der biometrischen Authentikation

Die folgende Spezifikation des Vorgangs der biometrischen Authentikation basiert darauf, dass sich unabhängig von verwendeter biometrischer Technik und dem Einsatzgebiet dieser Vorgang in vier Phasen gliedert. Ausgenommen von dieser Betrachtung ist die Merkmalsaufnahme, also die Generierung des Abbildes des biologischen Merkmals.

Die Phasen sind im Einzelnen:

1. Vorverarbeitung

   In dieser Phase finden alle Schritte statt, die für das weitere Vorgehen zwingend nötig sind, aber kein Bestandteil der eigentlichen Signaturberechnung sind. Die Rohdaten (wie z. B. die Bilddaten der Kamera oder ein Audio-Signals eines Mikrofons) werden untersucht und bearbeitet, um eine optimale Weiterverarbeitung zu ermöglichen. Die genaue Aufgabe der Vorverarbeitung hängt von der verwendeten biometrischen Technik und der Zielsetzung ab. Unter diesen Punkt fallen z. B. die Lokalisierung markanter Punkte bei der Gesichtserkennung oder die Hervorhebung der Muster eines Fingerabdrucks.

2. Qualitätssicherung und Normalisierung

   Im Anschluss an die Vorverarbeitung der Daten muss die Entscheidung getroffen werden, ob die Qualität des vorhandenen Datenmaterials ausreichend für eine erfolgreiche Signaturberechnung ist. Im Falle der Iriserkennung wird hier z. B. überprüft, ob das Abbild der Iris nicht zu stark von Lichtreflektionen und überdeckenden Augenlidern gestört ist. Falls die Datenqualität nicht ausreicht, bricht der Vorgang der Authentikation ab. Weiterhin finden Schritte der Normalisierung statt. Die Daten werden in ein definiertes Format gebracht, um so dem Vorgang der Signaturberechnung eine einheitliche Grundlage zu bieten, auf der die weiteren Berechnungen basieren.

3. Berechnung der biometrischen Signatur

   Diese Phase beinhaltet den eigentlichen Kern-Algorithmus der verwendeten biometrischen Technik. Aus den vorverarbeiteten Daten des Abbildes des biologischen Merkmals wird eine biometrische Signatur berechnet. Die Art der Berechnung und des Ergebnisses sind algorithmusabhängig.

4. Vergleich und Entscheidung

   In der abschließenden Phase findet der Vergleich der berechneten Signatur mit gespeicherten Signaturen aus der Datenbank statt. Dieses

beinhaltet im Falle der Identifikation (siehe Abschnitt 2.18) einen Vergleich mit allen Signaturen einer Datenbank oder im Falle der Verifikation den Vergleich mit einer ausgewählten Signatur[3]. Dieser Vergleich gibt ein Abstandsmaß an, auf Grund dessen der Grad der Ähnlichkeit zwischen dem aufgenommenen biologischen Merkmal und den gespeicherten biometrischen Signaturen festgelegt werden kann. Je nach Ergebnis der Vergleiche endet die Authentikation positiv (akzeptierend) oder negativ (zurückweisend).

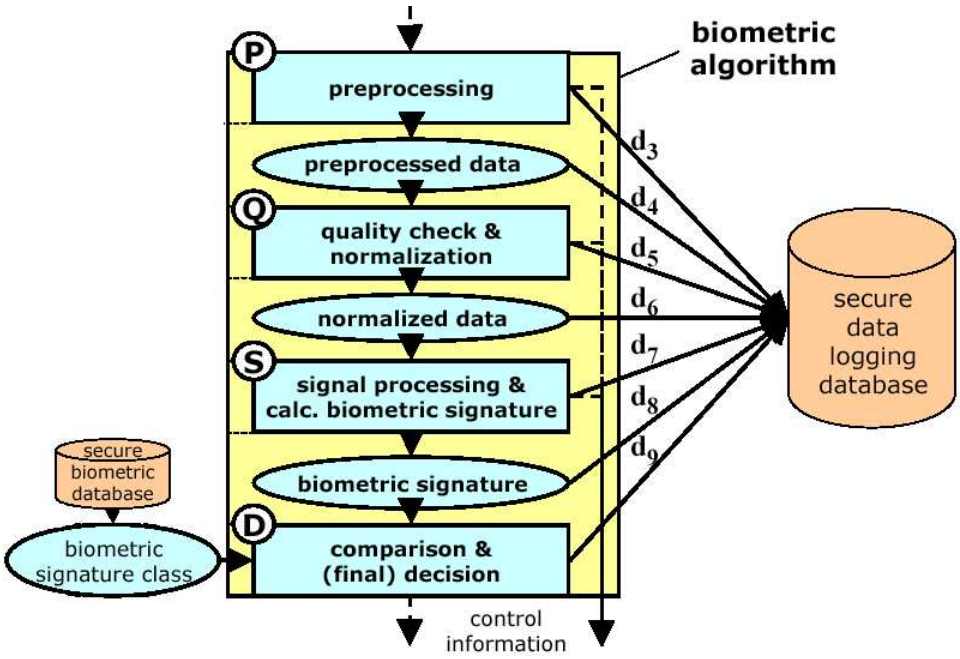


Abbildung 4.2: Der Vorgang der biometrischen Authentikation (mit Datensammlung) [Broe02b]

Den beschriebenen Phasen ist gemein, dass sie algorithmisch unabhängig sind. Jede der Phasen kann als einzelne Einheit betrachtet und behandelt werden. Dieser modulare Aufbau ermöglicht eine separate Entwicklung und Kalibrierung der einzelnen Phasen. So kann z. B. im Falle der Gesichtserkennung das Verfahren zum Auffinden des Gesichtes in den Rohdaten ausgewechselt werden, ohne notwendigerweise eine Anpassung der Methoden der anderen Phasen (wie der Signaturberechnung) zu bedingen[4].

[3] Dieses ist eine vereinfachte Darstellung. Es sind Verfahren denkbar, in denen auch im Falle der Verifikation Vergleiche mit einer Reihe von Signaturen vollzogen werden, z. B. um so Randbedingungen verschiedener Umgebungen ausgleichen zu können.

[4] Diese algorithmische Unabhängigkeit schließt aber keinesfalls aus, dass Erkenntnisse,

In vielen Fällen gliedern sich ebenfalls die Phasen selbst in separate Schritte, die eine unabhängige Modularisierung ermöglichen. Diese lassen sich aber nicht algorithmusübergreifend (also für verschiedene biometrische Techniken) verallgemeinern.

Wie in anderen Gebieten der Informatik ist auch in diesem Fall die Grundlage einer präzisen Modularisierung eine saubere Schnittstellendefinition.

Im weiteren Verlauf dieser Arbeit werde ich die oben beschriebene Modularisierung verwenden. Dabei tragen die Module folgende Bezeichnungen[5]:

1. P (*Preprocessing*): Vorverarbeitungsmodul
2. Q (*Quality Check & Normalization*): Normalisierungmodul
3. S (*Signature*): Modul zur Signaturberechnung
4. D (*Decision*): Entscheidungsmodul

Eine Anwendung dieser Modularisierung findet sich im folgenden Abschnitt.

4.3 Modularisierung des zu implementierenden Algorithmus

Eine Modularisierung des Algorithmus von John Daugman wurde wie folgt vorgenommen. Die hier nicht explizit angegebenen Schnittstellen der Module ergeben sich dabei direkt aus ihren Aufgaben.

4.3.1 Vorverarbeitung (P-Modul)

Das Vorverarbeitungsmodul erfüllt die Aufgabe des Findens der Iris in den Bilddaten und die Bestimmung der Grenzen zwischen Pupille/Iris und Iris/Sklera.

4.3.2 Qualitätssicherung und Normalisierung (Q-Modul)

Zum Zwecke der Normalisierung enthält das Q-Modul die Algorithmen zur Transformation des gefundenen Irisrings. Es kann argumentiert werden, dass die Transformation der Irisdaten ein Vorverarbeitungsschritt ist. Da sich

die aus Daten einer bestimmten Phase gewonnen wurden, nicht Konsequenzen auf die Anpassung einer anderen Phase haben können.

[5] Diese an [Broe02] angelehnte Namensgebung hat u.a. den Grund der Vereinheitlichung der Benennung vergleichbarer Module innerhalb verschiedener Arbeiten der Arbeitsgruppe Biometrik des Fachbereichs Informatik an der Universität Hamburg.

aber das Format der Daten, die an das S-Modul weitergereicht werden, zwingend aus dem Transformationschritt ableitet und das P-Modul keinen Einfluss auf das Datenformat haben sollte, ist eine Zuordnung der Routinen zum Q-Modul erforderlich. Weiterhin sind Implementierungen des Algorithmus möglich, die keine explizite, sondern eine implizite Verwendung des pseudo-polaren Koordinatensystems verwenden. In diesem Fall entfällt die Transformation zugunsten einer Initialisierung des Koordinatensystems, was eindeutigerweise ein Normalisierungsvorgang ist.

Im Rahmen der Qualitätssicherung findet eine Identifizierung und Lokalisierung der Störungen des Irisbildes (hervorgerufen z. B. von überdeckenden Augenliedern) statt. Darauf folgt eine Entscheidung, ob die Güte des Bildes für eine Signaturberechnung ausreichend ist. Im Falle einer positiven Entscheidung findet die Berechnung der Bitmasken statt, die für einen Ausschluss der gestörten Bereiche beim Signaturvergleich verantwortlich ist.

4.3.3 Signaturberechnung (S-Modul)

Das S-Modul beinhaltet sämtliche Algorithmen, die für die eigentliche Berechnung des Iriscodes zuständig sind. Diese umfassen im Besonderen die Routinen für die Wavelettransformation.

4.3.4 Vergleich (D-Modul)

Im D-Modul findet der Vergleich der berechneten biometrischen Signaturen mit den eingelernten Signaturen statt. In diesem Modul finden sich die Implementierungen der Vergleichs- und Entscheidungsroutinen (für z. B. die Berechnung des Hamming-Abstandes).

4.4 Einschränkung der Freiheitsgrade

Eine erfolgreiche Parametrisierung eines biometrischen Algorithmus ist kein triviales Vorhaben. Algorithmen, die auf der Verwendung von Wavelets basieren, besitzen eine große Anzahl von Freiheitsgrade. Vor dem Beginn der Implementierung bzw. der Experimente müssen diese identifiziert werden. Im Anschluss an die Bestimmung der Freiheitsgrade ist es nötig, eventuelle Beschränkungen in der Parametrisierung zu finden, um so den Aufwand der Versuche zu verringern. Diese Beschränkungen können sich sowohl aus dem Algorithmus selbst, insofern dieser bereits publiziert wurde, ergeben aus vorrausgehenden Überlegungen, welche die Natur der zu untersuchenden Daten in Betracht ziehen. In den folgenden Abschnitten soll am Beispiel des ausgewählten Algorithmus von J. Daugman das eben formulierte Vorhaben demonstriert werden.

Im Abschnitt 4.4.3, dessen Thema die im Rahmen dieser Arbeit vorgenommenen Schritte zur Einschränkung der Freiheitsgrade ist, werden die angewendeten Untersuchungen bzw. Überlegungen dokumentiert. Dabei findet in den meisten Fällen eine Einschränkung des Wertebereichs der Parameter für die, in Abschnitt 4.6 dokumentierten Experimente statt. Dieser Abschnitt soll dabei das Vorgehen demonstrieren und die Wahl der Wertebereiche erklären, beansprucht aber nicht mit den Einschränkungen in jedem Fall die beste Wahl getroffen zu haben. Zur Validierung einer derartigen Aussage sind Versuche nötig, deren Umfang über den Rahmen dieser Arbeit hinausgeht.

4.4.1 Identifizierung der Freiheitsgrade

In diesem Abschnitt sollen alle Freiheitsgrade des Algorithmus identifiziert werden, um so eine systematische Einschränkung dieser Freiheitsgrade zu ermöglichen.

Das Kernstück der Signaturgenerierung ist die Formel der Wavelettransformation:

$$h_{\{Re,Im\}} = sgn_{\{Re,Im\}} \int_{\rho} \int_{\phi} I(\rho,\phi)\ e^{-i\omega(\theta_0-\phi)} e^{-(\tau_0-\rho)^2/\alpha^2} e^{-(\theta_0-\phi)^2/\beta^2} \rho\ d\rho\ d\phi. \tag{4.1}$$

Eine genauere Untersuchung der Formel wurde in Abschnitt (3.1.1) vorgenommen. Aus der Formel ergeben sich folgende Freiheitsgrade:

1. Skalierung der Wavelets:

   Die Skalierung der Wavelets wird über die Parameter α und β bestimmt. β reguliert die Ausdehnung der Wavelets in radialer Richtung, α die Ausdehnung bezüglich der Grenze zwischen Pupille und Iris. Im transformierten Bild entspricht dies bezüglich β der vertikalen und bezüglich α der horizontalen Ausdehnung der Wavelets.

2. Ortskonstanten:

   Die Positionierung der Wavelets bezüglich der Irisdaten findet über die Parameter θ_0 und τ_0 statt. Abgebildet auf die transformierten Bilddaten bestimmt θ_0 die vertikale und τ_0 die horizontale Positionierung der Wavelets.

3. Frequenz:

   Die mittlere Frequenz des Gaborwavelets wird über den Parameter ω eingestellt. Für die konkrete Ausprägung des Wavelets ist das Verhältnis von ω zu α bestimmend. Je größer der Koeffizient ω/α ist, desto mehr Schwingungen beinhaltet das Gaborwavelet (siehe auch Abschnitt 4.4.3).

Bemerkung 4.4 (Richtung der Wavelets)
Im Gegensatz zu anderen Algorithmen, die Gaborwavelets verwenden, findet sich in der vorliegenden Formel kein Parameter, der die Ausrichtung des zweidimensionalen Wavelets bestimmt.

4.4.2 Algorithmus-inhärente Einschränkungen

In den folgenden Abschnitten werden die Niederschriften des Algorithmus von John Daugman detailliert dahingehend untersucht, inwieweit sie Anhaltspunkte für eine Parametrisierung enthalten. Die auf diese Weise gewonnenen Informationen sollen die Grundlage für weitere Überegungen und Versuche bilden.

Für diese Untersuchung hat es sich als hilfreich erwiesen, dass John Daugman seinen Algorithmus im Verlauf der letzten zehn Jahre wiederholt publiziert hat. Z. B. gibt die Antragsschrift auf die Erteilung des Patents [Patent] relativ genaue Angaben zur Positionierung der untersuchten Punkte der Iris, enthält aber wenig Informationen über die Anzahl und Ausprägung der verwendeten mittleren Frequenzen. Zu diesem Aspekt lassen sich Anhaltspunkte in dem Artikel [Daug01] finden. Die detaillierteste Beschreibung über das Vorgehen der Transformation der Bildpunkte in das dimensionslose Koordinatensystem enthält [Daug93]. Exakte Angaben sind aber erwartungsgemäß in keiner der Publikationen zu finden.

Anzahl der zu untersuchenden Punkte

Der von Daugman berechnete Iriscode hat eine Größe von 2048 Bit. Wie in Abschnitt (3.1.1) erläutert, werden pro Berechnung eines Waveletkoeffizienten zwei Bit dieses Codes gesetzt, es werden also insgesamt 1024 Waveletkoeffizienten berechnet.

Positionen der Analyse-Bänder

Wie schon erläutert, positionieren sich die untersuchten Punkte der Iris entlang mehrerer konzentrischer *Analyse-Bänder* („Analysis Bands“). In Abschnitt (3.1.1) wurde gezeigt, wie die Radien der Grenzen des Irisrings ermittelt werden. Basierend auf diesen Radien, werden die Radien der Analyse-Bänder festgelegt.

Aus [Patent, S. 9]:

- Die Größe des Radius des innersten Bands beträgt ungefähr das 1,1-fache des durchschnittlichen Pupillenradius, um so Unregelmäßigkeiten der Pupillengrenze auszugleichen und um sicherzustellen, dass die Pupille vollständig von dem Gebiet der Analyse ausgeschlossen ist.

- Das äußerste Band wird bei 80% des radialen Abstandes zwischen den Grenzen Pupille/Iris und Iris/Sklera angesetzt. Diesen verhältnismäßig großen Sicherheitsabstand erklärt Daugman mit der oft nicht kreisförmigen Gestalt der Iris.
- Von dem äußersten Band ausgehend, werden mit festgelegtem radialen Abstand insgesamt acht Ringe auf der Iris definiert.

Über die genaue Beschaffenheit der radialen Abstände der Bänder werden keine Angaben gemacht.

Skalierung und Frequenz der Wavelets

In [Patent, S. 10] gibt Daugman an, dass sich die Skalierungsparameter α und β umgekehrt proportional zu der Frequenz ω verhalten. Insgesamt treten acht verschiedene Skalierungen der Wavelets auf.

Das Verhältnis der Werte zueinander ist derart gewählt, dass alle verwendeten Wavelets selbstähnlich sind, sie also eine selbstähnliche Familie von „Frequency Selective Quadrature Filters“ [Patent] bilden.

> „They are self-similar, because the inverse proportionality of their size and frequency parameters renders them all dilates of each other sharing a common shape“ [Patent].

In [Daug01, S. 1738] gibt Daugman an, dass die Werte des Frequenzparameters ω einen Bereich von drei Oktaven[6] umfassen.

Zusammenfassung der angegebenen Einschränkungen

Eine Zusammenführung der oben genannten Ergebnisse führt zu folgenden Feststellungen:

- 1024 untersuchte Punkte, verteilt auf
  - acht Skalierungen und
  - acht Analyse-Bänder

Folgende Punkte bleiben nach dem Studium der Artikel von Daugman offen:

- Der exakte Frequenzbereich:

  Es steht lediglich die Information zur Verfügung, dass sich dieser Bereich über drei Oktaven erstreckt.

[6] Die von Daugman verwendete Notation bezieht sich auf eine logarithmische Skala zu der Basis zwei.

- Positionierung der Analyse-Bänder:

  Der radiale Abstand der Bänder zueinander ist nicht bekannt. Ob der naheliegende Ansatz optimal ist, die Bänder gleichmäßig über den Irisring zu verteilen, ist unbekannt. In eine erfolgreiche Positionierung der Bänder fließen Eigenschaften wie z. B. das Verhalten der Irismuster bei Pupillendilatation und die radiale Informationsdichte der Irismuster ein.

- Verteilung der zu untersuchenden Punkte:

  Bei der Verteilung von 1024 Punkten auf acht Skalierungen der Wavelets und acht Analyse-Bänder ergibt sich eine durchschnittliche Anzahl von 16 Punkten pro Band, die jeweils mit den skalierten Versionen des Wavelets untersucht werden. Es bleibt zu überlegen, ob diese kanonische Verteilung die sinnvollste ist. Äußere Bänder haben einen größeren Umfang als weiter zum Pupillenmittelpunkt positionierte Bänder. Eine Verwendung der gleichen Anzahl von untersuchten Punkte pro Band könnte also zu einem Informationsverlust bei der Untersuchung der äußeren Regionen der Iris führen. Außerdem sind Ansätze aus der Multiskalen-Analyse in Betracht zu ziehen, bei denen die Anzahl der untersuchten Punkte von der Skalierung der Wavelets abhängig ist: Viele Punkte sind mit „kleinen“ Wavelets und wenige Punkte mit „großen“ Wavelets zu untersuchen.

- Skalierung der Wavelets:

  In den Ausführungen Daugmans finden sich wenig Informationen zur Skalierung der Wavelets. Bekannt ist, dass das Verhältnis der horizontalen Skalierung (bestimmt durch den Parameter α) in einem definierten Verhältnis zu der dominierenden Frequenz (ω) steht, so dass sich eine Familie von selbstähnlichen Wavelets ergibt. Genaue Informationen über das Verhältnis von ω zu α stehen nicht zur Verfügung. Über die vertikale Skalierung der Wavelets (bestimmt über den Parameter β) macht Daugman keine Angaben.

Zusammenfassend kann gesagt werden, dass die Artikel, die im Zusammenhang mit dem Algorithmus publiziert wurden, wertvolle Hinweise bieten, aber weitere Überlegungen und Untersuchungen für eine Parametrisierung eines Algorithmus notwendig sind.

4.4.3 Weitere Einschränkung der Freiheitsgrade

Für weitere Einschränkungen der Freiheitsgrade dienen sowohl theoretische als auch praktische Mittel. Ausgehend von dem Vorwissen, das aus den Artikeln Daugmans resultiert, wurde untersucht, wie der Wertebereich der vorhandenen Unbekannten möglichst stark zu beschränken ist. Dabei hat es sich

als hilfreich erwiesen, Teile des Algorithmus (insbesondere das Vorverarbeitungsmodul) zu implementieren, um diese als Basis für Untersuchungen der Eigenschaften der Bilddaten zu verwenden.

Der Prozess der durchgeführten Einschränkung der Freiheitsgrade kann somit bereits als Teil der in 4.2.2 beschriebenen iterativen Anpassung der Parametrisierung und Implementierung betrachtet werden. Die vorgenommenen Versuche lieferten wertvolle Hinweise, die ihrerseits wieder zu Anpassungen der Implementierung führten.

Die im folgenden getroffenen Einschränkungen der Wertebereiche der Parameter bilden die Grundlage für die Experimente in Abschnitt 4.6.

Mittlere Frequenzen der Gaborwavelets

Wie oben bereits angegeben, besteht zwischen der horizontalen Skalierung und der mittleren Frequenz der Wavelets ein umgekehrt proportionales Verhältnis, welches dafür verantwortlich ist, dass die Wavelets verschiedener Skalierung eine selbstähnliche Familie bilden. Genaue Angaben zu dem Verhältnis werden von Daugman nicht gemacht.

Untersuchungen haben ergeben, dass die Bedingung der Bildung einer selbstähnlichen Familie erfüllt ist, wenn ein festes Verhältnis gewählt wird, derart dass das Produkt

$$\omega \cdot \alpha = k$$

konstant bleibt (siehe Abb. 4.7 und 4.6).

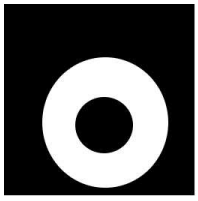


Abbildung 4.3: Testbild zur Untersuchung der DC-Sensitivität

Versuche mit Gaborwavelets mit unterschiedlichen Parametrisierungen von α und ω auf einfachen Bilddaten (siehe Abb. 4.3) zeigten, dass die Ausprägung der DC-Sensitivität (siehe Abschnitt 2.5) vom Verhältnis der beiden Parameter zueinander abhängt. Ein Vergleich von Abb. 4.4 zu Abb. 4.5 visualisiert diese Abhängigkeit:

1. Abb. 4.4 zeigt das Ergebnis einer Transformation des Testbildes mit Gaborwavelets mit einem Produkt von $\omega \cdot \alpha = 4$. Der Unterschied der

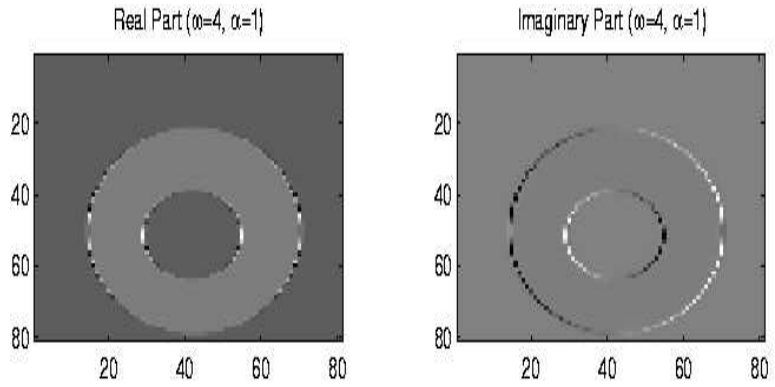


Abbildung 4.4: DC-Sensitivität bei einem Verhältnis von $\omega \cdot \alpha = 4$

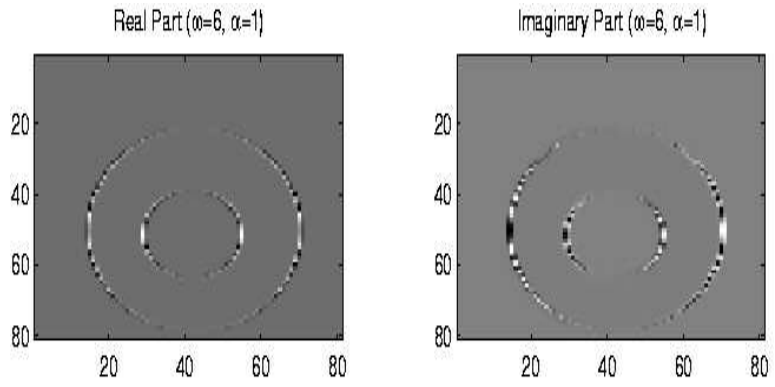


Abbildung 4.5: DC-Sensitivität bei einem Verhältnis $\omega \cdot \alpha = 6$

Graustufen des Bereichs innerhalb und außerhalb des Rings zeigt, dass die Filterantwort des Realteils der Wavelets bezüglich des konstant hellen Bildbereichs ungleich Null ist.

2. Abb. 4.5 zeigt das Ergebnis einer Transformation des Testbildes mit Gaborwavelets mit einem Produkt von $\omega \cdot \alpha = 6$. Es ist zu erkennen, dass der Realteil dieser Wavelets weniger stark DC-sensitiv ist; Die Grauwerte inner- und außerhalb des Ringes gleichen sich[7].

Für eine Feststellung der Auswirkung des Verhältnisses von ω zu α auf die Güte des Algorithmus wurden für die Experimente zwei Parametrisierungen gewählt:

[7] An dieser Stelle bietet es sich an, bei Versuchen der Weiterentwicklung des Algorithmus mit Ausgleichsgliedern zu experimentieren, wie es z. B. [Fröh00] (siehe Abschnitt 2.5) angibt.

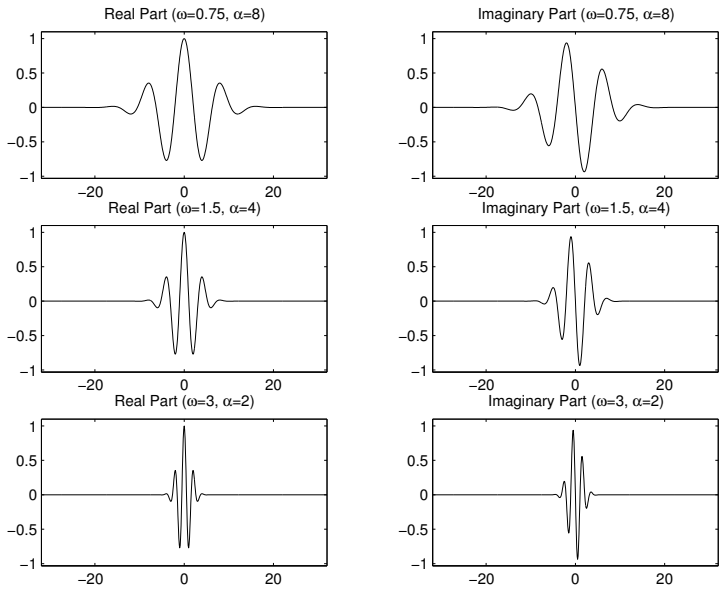


Abbildung 4.6: Gaborwaveletfamilie mit einem konstanten Produkt von $\omega \cdot \alpha = 6$ (eindimensionale Darstellung)

1. Die Gaborfamilie mit einem konstanten Produkt von $\omega \cdot \alpha = 6$, da diese im Rahmen der Versuche die geringste DC-Sensitivität aufwies.
2. Die Gaborfamilie, bei der das konstante Produkt 4 beträgt, da bei dieser der Realteil eine gute Approximation des „Mexican Hat“-Wavelets, ein in anderen Anwendungen oft verwendeter Ansatz, ist.

Die Abbildungen 4.7 und 4.6 zeigen eindimensionale Versionen dieser Familien.

Neben dem Verhältnis von Skalierung zu mittlerer Frequenz der Gaborwavelets ist auch der genaue Frequenzbereich unbekannt. Durch die nahe Verwandtschaft der Gabortransformation zur (gefensterten) Fouriertransformation bietet sich das Mittel der diskreten Fouriertransformation (DFT, siehe Bemerkung 2.2) zur Bestimmung der charakteristischen Frequenzen der Bilddaten an. Eine DFT der untransformierten Bilddaten zeigt eine Dominanz von niedrigen und mittleren Frequenzen aller Richtungen (siehe Abb. 4.8). Von größerer Aussagekraft ist die DFT von bereits transformierten Irisdaten (siehe Abb. 4.9): Die Transformation führt zu einer Konzentration auf horizontale Frequenzen. Diese treten über das gesamte Frequenzspektrum auf.

Aus dieser Erkenntnis lässt sich die Entscheidung Daugmans validieren, auf einen Parameter für die räumliche Ausrichtung der Wavelets zu verzichten.

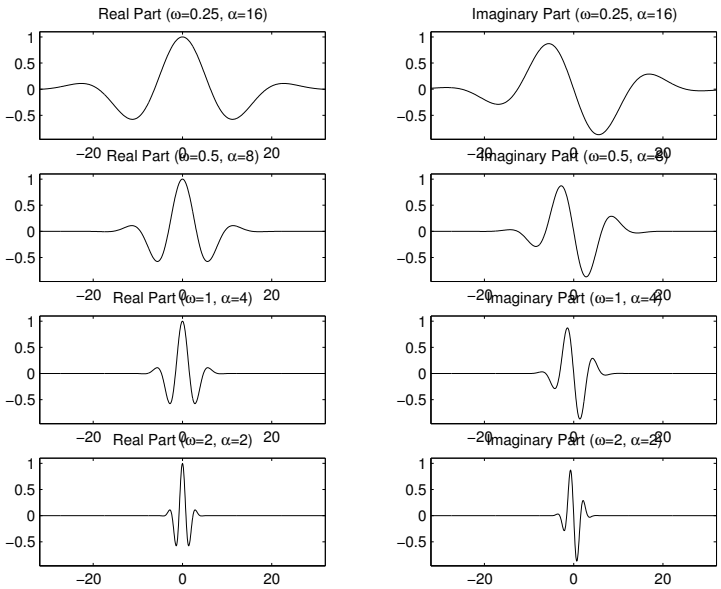


Abbildung 4.7: Gaborwaveletfamilie mit einem konstanten Produkt von $\omega \cdot \alpha = 4$ (eindimensionale Darstellung)

Signifikante Frequenzen treten nach der pseudo-polaren Transformation lediglich in horizontaler Richtung auf. Deswegen kann auf einer Untersuchung der Bilddaten mit Wavelets anderer Ausrichtungen verzichtet werden. Da bei den transformierten Daten horizontale Frequenzen über das gesamte Frequenzspektrum auftreten, kann eine Einschränkung des Frequenzbereichs vorerst nicht vorgenommen werden.

Bei der Vergabe von absoluten Werten für die Frequenzen ist die Abhängigkeit vom Datenformat der Vorverarbeitung zu betrachten (siehe hierzu Bemerkung 4.5).

Skalierung der Gaborwavelets

Die von Daugman verwendeten zweidimensionalen Gaborwavelets besitzen zwei Parameter zur Festlegung der Skalierung, deren Belegung sich umgekehrt proportional zu der mittleren Frequenz ω verhält. Die Ausprägung des horizontalen Skalierungsparameters α ist eng mit der Wahl der Frequenz verbunden, wie bereits im letzten Abschnitt ausgeführt wurde. Der vertikale Skalierungsparameter β hat einen vergleichsweise geringen Einfluss auf die Charakteristik des Wavelets, da dieser Parameter lediglich die vertikale Ausbreitung des Wavelets bezüglich der Irisdaten bestimmt, aber keinen Einfluss auf die Wellenstruktur des Wavelets besitzt.

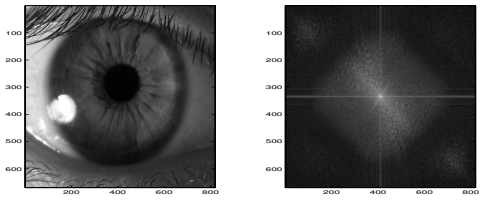


Abbildung 4.8: DFT von Irisdaten

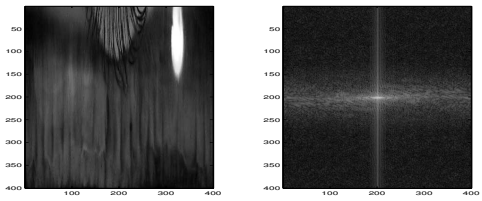


Abbildung 4.9: DFT von transformierten Irisdaten

Das Verhältnis von ω zu β ist, ebenso wie das Verhältnis von α zu ω, als umgekehrt proportional festgelegt; daraus folgt, dass das Verhältnis von α zu β konstant sein muss. Abgesehen von dieser Bedingung existieren keine weiteren Einschränkungen bezüglich der Parametrisierung von β.

Die Festlegung des Verhältnisses von α und β bestimmt somit effektiv die Maße des vertikalen Bereichs der Irisdaten bezüglich eines Analysepunkts, der in die Berechnung des Iriscodes einfließt.

Wählt man das Verhältnis $\frac{\alpha}{\beta}$ klein, ist die effektive vertikale Ausbreitung des Wavelets bezüglich der Irisdaten groß und es können sich, besonders bei niedrigen Frequenzen die Bereiche der Irisdaten, die ein Wavelet abdeckt, überschneiden. Dies führt dazu, dass es bei den betroffenen Waveletkoeffizienten zu Informationsredundanz kommen kann, da dieselbe Region der Iris auf mehr als einen Bestandteil des Iriscodes Einfluss hat.

Wird das Verhältnis $\frac{\alpha}{\beta}$ groß gewählt, ist also die effektive vertikale Ausbreitung des Wavelets klein, kommt es dagegen, besonders bei hohen Frequenzen, zu einer starken örtlichen Einschränkung des Analysebereichs in der vertikalen Nachbarschaft des jeweiligen Analysepunkts. Daraus kann resultieren, dass Regionen der Iris nicht von hochfrequenten Wavelets untersucht werden, und so wertvolle Informationen für den Iriscodes verloren gehen. Weiterhin erfordert ein kleiner Analysebereich eine hohe Präzision in der Festlegung der Analysepunkte, damit gewährleistet werden kann, dass dieselben Bestandteile eines Iriscodes auch dieselben Regionen der Iris beschreiben.

Somit ist eine Wahl des Verhältnisses der vertikalen zu der horizontalen

Skalierung eines Wavelets immer mit einer Gewichtung verbunden, ob eine guten Abgrenzung der Analysebereiche der Wavelets niedriger Frequenz oder eine möglichst vollständige vertikale Abdeckung der Irisdaten mit Wavelets hoher Frequenz Priorität hat.

Bemerkung 4.5 (Abhängigkeit von der Vorverarbeitung)
Ein Verhältnis α zu β muss immer im Zusammenhang mit dem Datenformat der Irisdaten, das im Normalisierungsmodul generiert wird, betrachtet werden. Je nach dem, welches Verhältnis von horizontaler zu vertikaler Größe die transformierten Irisdaten haben, muss das Verhältnis $\frac{\alpha}{\beta}$ angepasst werden. Diese wird deutlich, wenn man zwei Transformationen gegenüberstellt, wobei die erste Transformation die Irisdaten (bezüglich des pseudo-polaren Koordinatensystems) in ein Bild von 512x100 Pixel transformiert, und die zweite als Transformationsergebnis ein Bild mit einer Auflösung von 400x400 Pixel erzeugt.

Dieses gilt ebenso für die Absolutwerte der gewählten Frequenzen. Diese sind stets relativ zu dem Datenformat zu sehen. So entspricht z. B. ein Wavelet mit einer Parametrisierung von ω gleich 2 bezüglich eines transformierten Bildes von 200x200 Pixel, einem Wavelet mit einer mittleren Frequenz $\omega = 1$ bezüglich eines transformierten Bildes von 400x400 Pixel.

Eine Festlegung des in den Versuchen verwendeten Verhältnisses $\frac{\alpha}{\beta}$ kann somit erst vorgenommen werden, wenn der Wertebereich von ω und das Datenformat der Vorverarbeitung bestimmt wurden.

Position der Analysepunkte

Bei der Festlegung der Position der Analysepunkte bieten sich zwei unterschiedliche Modelle der Verteilung an:

1. „Jet“-Ansatz

   Bei diesem Ansatz wird jeder Analysepunkte mit allen zur Verfügung stehenden Mitgliedern der Waveletfamilie untersucht. Das bedeutet in unserem Fall, dass insgesamt 128 Punkte auf die Analysebänder verteilt werden, die jeweils mit allen Wavelets der verwendeten acht Skalierungen untersucht werden. Dieses führt zu einer klaren Charakterisierung der festgelegten Punkte.

   Dieses Verfahren ist vergleichbar mit den in Abschnitt 3.3.1 beschriebenen Jets, allerdings mit dem Unterschied, dass bei den dort beschriebenen Jets zusätzlich Wavelets unterschiedlicher Ausrichtung Verwendung finden. Da im Abschnitt über die Einschränkung des Wertebereichs der mittleren Frequenz ω gezeigt wurde, dass Daugmans Beschränkung auf eine Ausrichtung der Gaborwavelets valide ist, führt

diese Abwandlung des Jet-Ansatzes zu keinem Widerspruch. Wie in Kapitel 3 dargestellt, ist das Verfahren, Bildregionen über Jets von Gaborwavelets zu charakterisieren, verbreitet und etabliert.

2. Multiskalen-Ansatz

   Der zweite Ansatz, der sich anbietet, leitet sich aus den Erkenntnissen der diskreten Wavelettransformation ab (siehe Abschnitt 2.1.7): Anstatt jeden Punkt mit allen vorhandenen Frequenzen zu untersuchen, wird eine der Größe der Wavelets angepasste Verteilung der Punkte festgelegt. Da Wavelets von niedriger Frequenz eine größere räumliche Ausdehnung haben als hochfrequente, untersucht man nur wenige Punkte mit „großen“ Wavelets und viele Punkte mit „kleinen“. So kann eine Verteilung der Analysepunkte gefunden werden, durch die für jede Frequenz eine breite Abdeckung aller Regionen der Iris vorgenommen wird und eine Informationsredundanz durch sich überschneidende Wavelets vermieden wird.

   Es ist jedoch fraglich und bedarf weiterer Untersuchungen, ob die Erkenntnisse der Theorie der diskreten Wavelets unadaptiert für Gaborwavelets übernommen werden können.

Nach einer Entscheidung über die Zuordnung von Analysepunkten und Frequenzen muss weiterhin die allgemeine Verteilung der Analysepunkte auf die Analysebänder festgelegt werden. Naheliegend ist die kanonische Verteilung, also eine Verteilung, bei der jedes Analyseband dieselbe Anzahl von Analysepunkten beinhaltet. Die Struktur der untersuchten Bilddaten gibt jedoch Hinweise auf mögliche andere Verteilungen:

1. Auflösungsabhängige Verteilung

   Durch die ringartige Form der Irisdaten unterliegen die (radial gesehen) äußeren Strukturen einer höheren Auflösung als die Strukturen, die näher an der Pupille liegen. Diese Beobachtung deutet darauf hin, dass eine höhere Anzahl von Analysepunkten auf den äußeren Bändern zu besseren Ergebnissen führen könnte.

2. Strukturabhängige Verteilung

   Eine subjektive Betrachtung von Irisbildern führt zu der Beobachtung, dass bei den meisten Iriden die Muster in der Nähe der Pupille am detailliertesten sind. Diese Regionen verfügen somit über ein höheres Potential an Informationen zur Unterscheidung und Identifikation verschiedener Iriden. Diese Beobachtung legt nahe, dass eine genauere Analyse der inneren Regionen in einer verbesserten Erkennungsgenauigkeit resultieren könnte. Dieser Ansatz spricht für eine größere Anzahl von Analysepunkten auf den inneren Bändern.

Die beiden angegebenen Punkte bedürfen vor einer abschließenden Beurteilung weiterer Untersuchungen, die aus Aufwandsgründen nicht in dieser Arbeit vorgenommen werden können.

Die Versuche im Abschnitt 4.6 verwenden eine kanonische Verteilung der Punkte nach dem oben beschriebenen „Jet“-Ansatz, also 16 Analysepunkte pro Band, die jeweils mit allen acht Versionen des Gaborwavelets untersucht werden.

Transformation des Irisrings

Die Transformation der Irisdaten in das pseudo-polare Koordinatensystem ist kein Teil des Kern-Algorithmus. Somit betreffen die folgenden Betrachtungen den Wertebereich der identifizierten Freiheitsgrade nur indirekt. Dennoch sind vorausgehende Überlegungen zur Ausprägung der Transformation in diesem Abschnitt notwendig, da die Durchführung der Transformation unmittelbare Auswirkungen auf die Parametrisierung der Variablen in der Formel 4.1 haben (siehe hierzu auch Bemerkung 4.5).

Für jedes der verwendeten Irisbilder wurde jeweils der Radius der Pupille und der Radius der Iris ermittelt. Dabei ergab sich ein durchschnittlicher Pupillenradius von etwa 65 Pixel und ein durchschnittlicher Irisradius von etwa 230 Pixel. Aus diesen Werten ergibt sich[8] ein gerundeter Pupillenumfang von 400 Pixel und ein gerundeter Irisumfang von 1450 Pixel.

Aufgrund dieser Werte wurde für das Zieldatenformat der Transformation eine horizontale Auflösung von 800 Pixel (der Umfang des Irisbands, das sich in der Mitte zwischen Pupillen- und Sklerarand befindet) und eine vertikale Auflösung von 200 Pixel (dieser etwas über dem durchschnittlichen Betrag[9] liegende Wert soll einem Verlust von vertikalen bzw. radialen Details vorbeugen).

4.5 Implementierung

Die folgenden Abschnitten enthalten eine Beschreibung der angefertigten Implementierung statt. Im Vordergrund steht bei dieser Beschreibung nicht die Dokumentation des Quellcodes, sondern die Spezifikation des Leistungsumfangs und die Erläuterung der Beschränkungen der Implementierung.

4.5.1 Auswahl der Entwicklungsumgebung

Als Entwicklungumgebung für die Versuche wurde die Kombination aus dem mathematischen Programmier- und Modellierungswerkzeug Matlab 6 (Re-

[8] Unter der Verwendung der Formel des Kreisumfangs: $U = 2r \cdot \pi$.

[9] Bezogen auf die Distanz zwischen Pupillen- und Sklerarand.

lease 12.1)[10] und der Programmiersprache C++ gewählt. Matlab ist eine integrierte Programmier- und Testumgebung, die sowohl die Durchführung interaktiver Versuche, wie auch die Ausführung erstellter Programme unterstützt. Der Name „Matlab“ leitet sich aus dem Begriff der Matrix ab: Alle von Matlab verwendeten Datentypen basieren auf Vektoren oder Matrizen. Zur Programmierung stellt Matlab eine imperative Script-Sprache, den Matlab-Code, zur Verfügung. Neben vielseitigen Bibliotheken für verschiedene Bereiche der Mathematik (die so genannten „Toolboxes“) existieren Routinen zur Darstellung digitaler Bilder und zur Visualisierung von Daten.

Matlab erlaubt die Auslagerung von Programmteilen als C(++)- oder Fortran-Routinen. Bei der Integration von C++-Routinen in Matlab kommt der spezielle C++-Compiler „mex“ zum Einsatz, der die C++-Routinen mit einer Schnittstelle versieht, die einen direkten Aufruf des Compilats aus Matlab heraus ermöglicht [Math].

Als Entwicklungsrechner diente eine SunBlade 100 mit 256 MB Hauptspeicher und dem Betriebssystem Solaris 8.

Gründe für die Wahl von Matlab

Die Gründe für die Wahl von Matlab gegenüber einer Implementierung in C++ ergaben sich aus den folgenden Punkten:

- Eine eigene Erstellung von Routinen zur graphischen Darstellung von Bildern und Funktionsdaten war im Zeitrahmen dieser Arbeit nicht durchführbar. Für eine effiziente Kontrolle der Korrektheit der Implementierung und für die Auswertung der Ergebnisse ist eine visuelle Darstellung jedoch unverzichtbar. Matlab stellt diese Routinen zur Verfügung.

- Matlab bietet umfangreiche mathematische Bibliotheken, die eine Implementierung vereinfachen und beschleunigen.

- Die interaktive Oberfläche, auf der einzelne mathematische Schritte ohne Programmieraufwand durchgeführt werden können, erlaubt ein schnelles Testen von Programmentwürfen.

- Die imperative Script-Sprache von Matlab ermöglicht eine vereinfachte Übertragung des erstellten Codes in Implementierungen anderer imperativer Programmiersprachen, speziell C++. So konnten sämtliche in C++ ausgelagerte Routinen zuvor prototypisch in Matlab-Code erstellt und getestet werden, bevor sie in C++ übertragen wurden.

[10] Siehe http://www.mathworks.com

Als Alternative zu Matlab hat sich eine Auswahl von „Mathematica“ herrausgestellt (ein vergleichbares Werkzeug, das z. B. in [Schr00] verwendet wird). Mathematica besitzt im Vergleich zu Matlab eine bessere Unterstützung von kontinuierlichen Funktionen. Das Fehlen von Routinen zur einfachen Darstellung digitaler Bilddaten (wie die Irisdaten) und die von Mathematica verwendete funktionale Programmiersprache, die eine Portierung des erstellten Codes in C++ erschwert, haben jedoch zu einer negativen Entscheidung gegenüber Mathematica geführt.

Ein weiterer Grund für die Wahl von Matlab war die Existenz der „Wavelettoolbox“, eine Sammlung von Routinen und Daten speziell für die Verwendung von Wavelets in Matlab, die sich für anfängliche Experimente mit Wavelets als gut geeignet erwies. Für die eigentliche Implementierung konnte die Toolbox leider nicht verwendet werden, da Gaborwavelets nicht Teil des Pakets waren und die vorhandenen Routinen für zweidimensionale Wavelettransformationen zu unflexible Schnittstellen aufwiesen.

Eine der beiden Hauptquellen für den Abschnitt 2.1.4 über die Grundlagen der Wavelets [Bäni01] enthält textbegleitend Matlab-Code, was sowohl die Einarbeitung in Matlab, als auch das Erlernen der Waveletgrundlagen erleichtert hat. Nicht zuletzt Grund für die Entscheidung Matlab zu verwenden, war die Information einer Mitarbeiterin von Iridian, der Firma der kommerziellen Implementierung des Daugman-Verfahrens, dass Iridian ebenfalls Matlab zur Entwicklung ihrer Algorithmen einsetzt.

Nachteile der Wahl

Ein Problem, dass sich aus der Verwendung von Matlab ergab, ergab sich auf die Spezialisierung des Programms auf matrixverwandte Bereiche der Mathematik: Mathematische Verfahren, deren Grundlage der abstrakte Funktionenbegriff ist (wie z. B. die Differenzierung oder Integration), sind z.T. schlecht in das Grundparadigma von Matlab („alle Daten sind Matrizen“) zu integrieren. Die Verfügbarkeit und Effizienz der Routinen zur numerischen Integration über mehrere Dimensionen, wie sie für die Wavelettransformation benötigt werden, ist beschränkt.

Aus diesem Grunde wurden die Routinen für die eigentliche Wavelettransformation von Grund auf in C++ implementiert[11].

[11] Im Verlauf der Implementierung wurde der Einsatz einiger frei erhältlicher C(++)-Bibliotheken zur zweidimensionalen Integration getestet. Diese Versuche scheiterten aus verschiedenen Gründen: Die verwendete Version der Bibliothek [Robi02] verursachte im Zusammenspiel mit Matlab so erhebliche Rundungsfehler, dass diese das Ergebnis verzerrten; bei der Bibliothek [Cool97] scheiterte eine Integration in Matlab, da eine Compilation der Bibliothek mit dem von Matlab verwendeten C++-Compiler nicht möglich war.

4.5.2 Umfang der Implementierung

Die angefertigte Implementierung dient dem Zweck der praktischen Umsetzung der betrachteten Routinen zur Wavelettransformation und der beispielhaften Durchführung einer iterativen Verfeinerung der Parametrierung. Ein vollständiger Nachbau eines alleinstehenden Iriserkennungssystems war nicht beabsichtigt. Aus diesem Grund wurden sämtliche Module, deren algorithmische Grundlagen nicht in der Wavelettheorie beheimatet sind, dieser Aufgabenstellung angepasst. Dieses umfasst besonders die Bereiche der genauen, automatischen Lokalisierung der Pupillen- und Irisgrenzen und die Bestimmung der gestörten Bildbereiche. Implementierungen dieser Teilaspekte sind eigenständige Aufgaben, die mit dem Thema dieser Arbeit nur wenig Berührungspunkte besitzen.

Das Modul zum Vergleich der Iriscodes wurde im Vergleich zu den Spezifikationen Daugmans erweitert, um so genauere Informationen zu gewinnen (so wird z. B. eine frequenzabhängige Aufspaltung des Hamming-Abstandes vorgenommen).

4.5.3 Implementierung der Module

Die Grundlage für die Struktur der Implementierung bildete die in Abschnitt 4.3 vorgenommene Modularisierung, deren Inhalt im weiteren als bekannt vorausgesetzt wird.

Für die einzelnen Module wird der implementierte Leistungsumfang und bei Bedarf die verwendete Methode beschrieben. Wenn nicht anders angegeben, wurden die Module in Matlab-Code verfasst. Eine Zusammenstellung der erstellten Routinen mit ihren Schnittstellen findet sich in Anhang C.

P-Modul

Als erster Vorverarbeitungsschritt wird das Bild in ein Graustufenbild umgewandelt. Dieses ermöglicht eine Weiterverarbeitung als MxN-Matrix. Eine Lokalisierung der Irisgrenzen wurde, wie bereits erwähnt, nicht implementiert. Stattdessen wurde manuell für jedes der verwendeten Irisbilder eine Datei mit Metadaten angelegt. Diese Datei enthält:

- die Koordinaten des Pupillenmittelpunkts,
- die Koordinaten eines Punktes auf der Grenze Pupille/Iris,
- die Koordinaten eines Punktes auf der Grenze Iris/Sklera, dessen vertikale Lage dem Pupillenmittelpunkt entspricht und dessen horizontale Position sich im linken Bildbereich befindet,

- die Koordinaten eines Punktes auf der Grenze Iris/Sklera, dessen vertikale Lage dem Pupillenmittelpunkt entspricht und dessen horizontale Position sich im rechten Bildbereich befindet.

Aus diesen Koordinaten werden der Pupillenradius und zwei Radien für die äußere Grenze der Iris berechnet. Die Erfassung zweier Irisradien ist nötig, da in den meisten Fällen[12] die Pupille nicht zentrisch in der Iris positioniert ist. Anhand der zwei Irisradien kann der elliptische Verlauf der Irisgrenze approximiert werden (siehe Abb. 4.10).

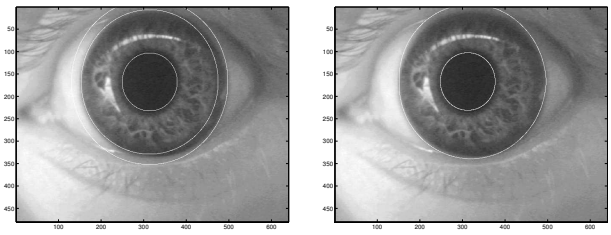


Abbildung 4.10: Visualisierng der Bestimmung der Grenze zwischen Iris/Sklera über zwei Radien

Q-Modul

Anhand der drei im P-Modul berechneten Radien wird der Irisring in eine rechtwinklige Datenrepräsentation (die verwendete Instanziierung des pseudo-polaren Koordinatensystems) transformiert.

Von einer Reihe von Ellipsen, deren Schwerpunkt der Pupillenmittelpunkt ist, werden jeweils die gleiche Anzahl von Punkten übernommen. Der Radius der Ellipsen wächst dabei gleichmäßig. In den hier durchgeführten Versuchen entspricht dieses einer Verwendung von 200 Ellipsen mit wachsendem Radius (wobei die erste Ellipse der Grenze Pupille/Iris und die letzte der Grenze Iris/Sklera entspricht), derem Rand jeweils 800 äquidistante Punkte entnommen werden. Dies führt, unabhängig vom Datenformat des Bildes und der relativen Größe der Irisdaten innerhalb des Bildes, zu einem einheitlichen Bildformat von 800x200 Pixel, das an die anschließenden Routinen weitergegeben wird.

Methoden zur Qualitätssicherung oder Feststellung von gestörten Bildbereichen wurden nicht erstellt.

[12] Soweit aus dem vorhandenen Bildmaterial verallgemeinert werden kann.

S-Modul

Der Kern dieses Moduls ist die eigentliche Wavelettransformation mit Gaborwavelets. Eine Beschreibung der Implementierung des Moduls lässt sich zweiteilen: in die Beschreibung der Routine zur Berechnung der Waveletkoeffizienten und in die Beschreibung der übergeordneten Routine, welche die Belegung der Übergabeparameter bestimmt und die Berechnung steuert.

Die Routine zur Berechnung der Waveletkoeffizienten wurde aus Performanzgründen in C++ entwickelt. Ein in Matlab-Code erstellter Prototyp der Routinen hatte ein Laufzeitverhalten, das eine praktische Anwendung wie in den vorgenommenen Versuchen unmöglich gemacht hätte[13]. Der Routine werden neben den Bild- und Waveletdaten (mittlere Frequenz, horizontale Ausbreitung, vertikale Ausbreitung) eine Reihe von Bildkoordinaten (in der Form zweier Vektoren) übergeben. Die Routine berechnet für jede dieser Koordinaten den entsprechenden Waveletkoeffizienten unter Verwendung der übergebenen Waveletdaten. Die Koeffizienten werden als Werte-Matrix zurückgegeben. Da C++ keine effiziente Art des Umgangs mit komplexen Zahlen kennt, werden Realteil und Imaginärteil der komplexen Koeffizienten in separaten Arbeitschritten berechnet.

Die Routine zur Steuerung der Berechnung erhält bei der Initialisierung Startwerte für die Waveletdaten (also für mittlere Frequenz und räumliche Ausbreitung), sowie die Anzahl der Berechnungsschritte und die zu verwendende Schrittgröße. Bei einer Schrittgröße von 1 finden die Berechnungsschritte bezüglich der mittleren Frequenz in Oktavenschritten statt. Bei einer Schrittzahl von 4, einer Schrittgröße von 1 und einem Startwert für die mittlere Frequenz von 2 findet somit beispielsweise eine Berechnung der Waveletkoeffizienten für die mittleren Frequenzen 2, 4, 8 und 16 statt. Wie in Abschnitt 4.4.3 schon erläutert, leiten sich die Belegungen der Parameter für die räumliche Ausbreitung der Wavelets direkt aus dem Frequenzparameter ab.

Abschließend wird aus den komplexen Waveletkoeffizienten, so wie es Daugman spezifiziert hat, der binäre Iriscode berechnet. Am Ende eines Durchlaufs, gibt das Modul nicht nur den ermittelten Iriscode zurück, sondern zusätzlich weitere Daten, die zur Kontrolle, Analyse und Ausgabe verwendet werden (dazu gehören u.a. die verwendeten Waveletdaten).

[13] Hiermit soll nicht ausgeschlossen werden, dass bei einer anderen Art der Programmierung des Matlab-Codes eine bessere Performanz zu erreichen wäre. Eine Abwägung des Zeit/Nutzen-Verhältnisses führte zu der Entscheidung, eine Optimierung des Matlab-Codes zu Gunsten einer C++-Implementierung zu verwerfen.

D-Modul

Wie oben bereits angesprochen, unterscheidet sich die Aufgabe dieses Moduls, in der angefertigten Implementierung, von der Aufgabe im Rahmen einer reinen Authentikations-Anwendung: Im Falle einer Authentikation muss eine berechnete Signatur gegen eine Referenzsignatur geprüft werden, um festzustellen, ob die Identität des Trägers des aufgenommenen Merkmals mit der zu der Referenz abgelegten, Identität übereinstimmt. In der angefertigten Implementierung ist die Aufgabe dieses Moduls die Ermittlung, ob die verwendete Parametrisierung für eine Anwendung in dem Algorithmus geeignet ist.

Der eigentlichen Berechnungen der Hamming-Abstände geht eine Sammlung der Signaturen voraus. Erst wenn für alle Irisbilder eines Testdatensatzes die Signaturen berechnet wurden, kommt es zu einem Vergleich[14]. Es werden die Hamming-Abstände innerhalb einer Klasse von Bildern (siehe 4.6.1) und zwischen den Klassen errechnet. Um eine genauere Analyse der Ergebnisse zu ermöglichen, findet weiterhin eine Aufspaltung der Ergebnisse nach den einzelnen mittleren Frequenzen der Wavelets statt. Diese wird durch Hamming-Abstände mit Bitmasken (siehe Formel (3.8)) realisiert, indem Bitmasken verwendet werden, die nur die Teile des Codes, die sich aus Wavelets der gewählten Frequenz ergeben haben, in die Berechnung des Hamming-Abstandes einfließen lassen.

4.5.4 Routinen zur Steuerung und Auswertung

Neben den Routinen, die nach den Modul-Spezifikationen erstellt wurden, sind auch Routinen zur Steuerung und Auswertung der Experimente Teil der Implementierung. Eine Beschreibung dieser Routinen findet sich in Anhang C.

4.6 Experimente

4.6.1 Vorbereitung der Experimente

1. Zusammenstellung von Testdatensätzen

Ein großes Problem, das sich im Rahmen des praktischen Teils dieser Arbeit ergab, war die mangelhafte Verfügbarkeit von Bildern der menschlichen Iris zu Testzwecken.

[14] Da bei fast jeder Versuchreihe unterschiedliche Parameter getestet werden, ist eine dauerhafte Speicherung der Signaturen von wenig Nutzen, da nur ein Vergleich von Signaturen mit exakt gleicher Parametrisierung sinnvoll ist.

Im Gegensatz zum Feld der Fingerabdruckerkennung, für das zu Testzwecken einige frei verfügbare Bibliotheken von Fingerabdruckbildern vorhanden sind, existiert ein vergleichbares Angebot für Irisbilder nicht. Erschwerend kommt hinzu, dass eine eigenständige Erstellung von Testbildern problematisch ist. In den frühen Phasen der Entwicklung eines Algorithmus zur Iriserkennung sind Bilder einer hohen Qualitätsstufe nötig, da jegliche Störungen Folgerungen aus den Testergebnissen erschweren.

Zum Test eines Algorithmus zur Iriserkennung ist eine Anzahl von Bildern derselben Iris nötig. Ausgehend von der angefertigten Implementierung, die keine Routinen zur Kompensation von Bildstörungen enthält (siehe 4.5.2), ergeben sich folgende Anforderungen an die zu verwendenden Sätze von Testdaten[15]:

- Mindestens drei unterschiedliche Bilder derselben Iris
- Keine großflächigen Lichtreflexionen
- Keine großflächigen Überdeckungen durch Augenlider
- Keine große Variation im Drehwinkel zur Kamera

Die Biometrikgruppe des Fachbereichs Informatik der Universität Hamburg hat mit verschiedenen Kameratechniken diverse Irisbilder aufgenommen, von denen aber leider nur ein geringer Anteil den formulierten Anforderungen genügten. Zusammen mit einer Reihe weiterer, in Qualität und Ausprägung verschiedener Irisbilder wurden zusammen mit diesen Bildern Testsätze von jeweils acht Bildern gebildet. Innerhalb dieser Testsätze wurden die Bilder in zwei Klassen gruppiert:

- Klasse 1

  Die Bilder in der Klasse 1 entstammen derselben Iris. Vergleiche der Signaturen innerhalb dieser Klasse geben Informationen, darüber, wie gut die gewählte Parametrisierung geeignet ist, Ähnlichkeiten zu erkennen.

- Klasse 2

  Die Bilder in der Klasse 2 entstammen menschlichen Iriden, deren Träger nicht mit dem Träger der Iris der Klasse 1 identisch ist. Vergleiche von Signaturen von Bildern der Klasse 2 mit Signaturen von Bildern der Klasse 1 geben Informationen, wie gut die gewählte Parametrisierung geeignet ist, unterschiedliche Iriden zu unterscheiden.

[15] Die Beurteilung der Erfüllung dieser Kriterien erfolgte rein subjektiv.

2. Festlegung des Verfahrens

Aus den in Abschnitt 4.4.3 vorgenommenen Untersuchungen ergab sich folgendes Verfahren: Innerhalb von vier Einzelschritten wird sukzessive ein geeigneter Wertebereich für die Parameter ermittelt. Zu diesem Zweck werden jeweils eine Reihe von alternativen Parametrisierungen festgelegt, die auf ihre Eignung getestet werden.

Die Schritte sind im Einzelnen:

1. Grobe Feststellung des Frequenzbereichs
2. Feine Feststellung des Frequenzbereichs
3. Untersuchung des Verhältnisses ω zu α
4. Untersuchung des Verhältnisse α zu β

In jedem dieser Schritte wird für jede gewählte Parametrisierung und jedes Irisbild des Testdatensatzes die biometrische Signatur berechnet. Für diese Signaturen werden anschließend die Hamming-Abstände der Bilder innerhalb der Klasse 1 und die Abstände zwischen den vorhandenen paarweisen Gruppierungen von je einem Bild der Klasse 1 und einem Bild der Klasse 2 berechnet.

4.6.2 Verlauf der Experimente

Angesichts der Tatsache, dass mit dem vorhandenen mangelhaften Bildmaterial keine weitgehend gültigen Ergebnisse erzielt werden können und eine vollständige Parametrisierung der Implementierung einen zu hohen Aufwand bedingt, der in den Zeitrahmen dieser Arbeit nicht zu integrieren war, wurde sich bei den vorgestellten Versuchen auf die exemplarische Demonstrationen des Vorgehens zur schrittweisen Eingrenzung des Wertebereichs der Parameter beschränkt. Für eine vollständige Parametrisierung ist, bei Beibehaltung der verwendeten Methoden, eine größere Anzahl von Abstufungen der Parameter (hier wurden zum Teil nur zwei alternative Belegungen verwendet) und eine erheblich größere Zahl von Irisbildern notwendig.

Bemerkung 4.6 (Die dokumentierten Testergebnisse)
Die im folgenden und im Anhang E beschriebenen Testergebnisse beziehen sich auf denselben, in Anhang D dokumentierten, Satz von Irisbildern. Diese Einschränkung wurde gewählt, um eine Vergleichbarkeit der Ergebnisse zu ermöglichen und die Dokumentation in einem übersichtlichen Rahmen zu halten[16].

[16] Im Rahmen der durchgeführten Experimente wurden weitere Testsätze von Irisbildern

Methode der Auswertung

Zur Auswertung wurden die Werte wie folgt verwendet. Für eine gute Parametrisierung ausschlaggebend ist einerseits die Distanz zwischen den mittleren Hamming-Abständen (innerhalb der Klasse 1 / zwischen den Klassen), da dieser Wert Auskunft darüber gibt, wie gut die Klassen von einander unterschieden werden, und der Distanz des maximalen Hamming-Abstandes innerhalb der Klasse 1 zu dem minimalen Hamming-Abstand zwischen den Klassen, da diese Distanz Informationen über die Güte der Parametrisierung bei Grenzfällen gibt (ist der minimale Hamming-Abstand zwischen den Klassen kleiner als der maximale Hamming-Abstand innerhalb der Klasse 1, gibt es mindestens eine Iris aus der Klasse 2, die als sehr ähnlich zu der Referenziris erkannt wird; eine zuverlässige Unterscheidung zwischen Klassen ist somit nicht mehr möglich).

Bemerkung 4.7 (Balkengrafiken)
Im weiteren Verlauf dieses Abschnitts, sowie in Anhang E mit Details zu den Versuchen, werden die Versuchsergebnisse in Form von Balkendiagrammen dargestellt. Diese Diagramme stellen in zwei separaten Grafiken eine Zusammenfassung der berechneten Hamming-Abstände dar. Die erste Grafik dient der schnellen Übersicht und stellt den mittleren Hamming-Abstand der Signaturen innerhalb der Klasse 1 dem mittleren Abstand der Signaturen zwischen den Klassen gegenüber. Die zweite Grafik enthält sechs Balken. Die ersten drei Balken beziehen sich auf die Hamming-Abstände der Signaturen innerhalb der Klasse 1. Die Balken stellen den mittleren, den minimalen und den maximalen Abstand dar. Die Gruppe der letzten drei Balken bezieht sich auf die Abstände der Signaturen zwischen den Klassen. Auch sie visualisieren den mittleren, minimalen und maximalen Hamming-Abstand.

1. Grobe Feststellung des Frequenzbereichs

Am Anfang erfolgt eine grobe Eingrenzung des Wertebereichs des Parameters ω, der die mittlere Frequenz der Wavelets festlegt. Dazu wurde eine Berechnung von Waveletkoeffizienten über den ganzen zur Verfügung stehenden Wertebereich von ω vorgenommen. Das Verhältnis von ω zu α wurde so festgelegt, dass sich ein Produkt von 4 ergab, d. h. der Realteil der Gaborwavelets ergibt also eine Approximation des „Mexican Hat“-Wavelets (siehe Abschnitt 4.4.3).

Bei einer Belegung von $\omega = 0.03125$ folgt eine Wert von 128 für α. Somit erstreckt sich das Wavelet über die gesamte Breite der transformierten Daten:

eingesetzt, mit z.T. vergleichbaren und z.T. unbrauchbaren Ergebnissen (hervorgerufen durch unzureichende Qualität der Bilder). Die Aussagekraft einer Einbeziehung dieser Ergebnisse ist, aus den oben erläuterten Gründen, zweifelhaft.

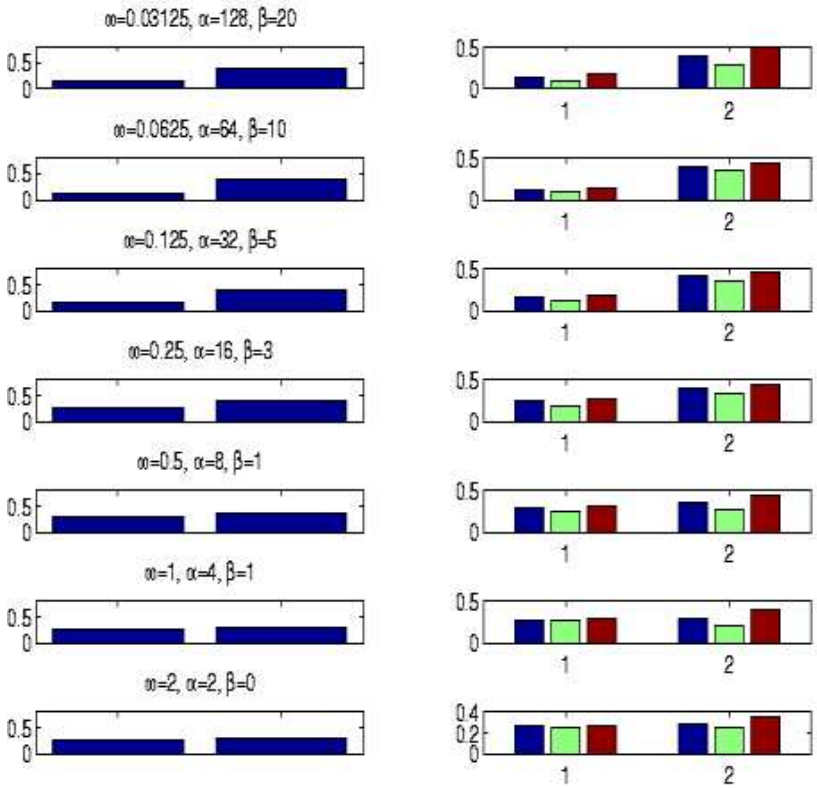


Abbildung 4.11: Ergebnis der groben Abschätzung des Wertebereichs für ω

Bei der verwendeten, der Formel Daugmans entnommenen, Implementierung der Gaborwavelets hat sich die Abschätzung als zutreffend erwiesen, dass das Produkt von α mit dem Wert 6 die ungefähre Abmessung der horizontalen Ausdehnung des Gaborwavelets in Pixeln ergibt. Bei einem Wert von $\alpha = 128$ hat das Gaborwavelet also eine horizontale Abmessung von ungefähr 768 Pixel (bei einer Auflösung der transformierten Irisdaten von 800x200 Pixel). Die vertikale Ausdehnung der Wavelets berechnet sich über die Belegung von β analog.

Für eine maximale Belegung von ω wurde der Wert 2 gewählt, da in vorausgehenden Experimenten die Erfahrung gemacht wurde, dass bei zu kleinen Wavelets keine verwertbaren Ergebnisse zu erzielen sind.

Daraus ergab sich ein Wertespektrum für ω von 0.03125 bis 2. Dieses ergibt einen Bereich von sieben Oktaven.

Wie Abb. 4.11 sowie den Einzelergebnissen der Berechnungen (siehe Anhang E.1) entnommen werden kann, nimmt der Unterschied der Hamming-Abstände von Bildern derselben Iris und Bildern unterschiedlicher Iriden mit sinkendem ω zu.

2. Feine Feststellung des Frequenzbereichs

Wie in Abschnitt 4.4.2 beschrieben, spezifiziert Daugman die Werte von ω auf acht verschiedene Belegungen, die einen Bereich von drei Oktaven abdecken.

Nachdem die grobe Abschätzung des Wertebereichs gezeigt hat, dass „große“ Wavelets zu anscheinend besseren Ergebnissen führen, wurde dieses Ergebnis als Grundlage für die Festlegung der Wertebereiche der folgenden Versuche verwendet:

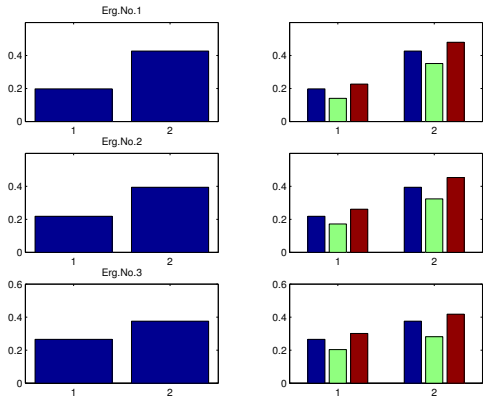


Abbildung 4.12: Ergebnis der feinen Abschätzung des Wertebereichs für ω

Es wurden insgesamt drei Versuchsreihen mit folgenden Belegungen unternommen:

1.	$\omega =$	0.03125	,	$\alpha =$	128	,	$\beta =$	20
2.	$\omega =$	0.04166	,	$\alpha =$	96	,	$\beta =$	15
3.	$\omega =$	0.0625	,	$\alpha =$	64	,	$\beta =$	10

Aus diesen Anfangswerten wurden Belegungen für ω, α und β berechnet, die, in gleichmässigen Abständen zueinander, eine Abdeckung von ω von drei Oktaven ergeben (Dieses gilt ebenso für die folgenden Versuche).

Wie aus Abb. 4.12 und den Einzelergebnissen der Berechnungen (siehe Anhang E.2) abgelesen werden kann, ergibt sich wieder eine Tendenz zu Wavelets großer Ausdehnung. Bei den Versuchreihen ergaben sich folgende Differenzwerte für die betrachteten Hamming-Abstände:

	Mittelwerte	Extremwerte
1.	0,229166	0,125000
2.	0,175521	0,062500
3.	0,109635	-0,019531

Die Spalte mit der Bezeichnung „Mittelwerte“ enthält die Differenz der mittleren Hamming-Abstände (innerhalb der Klasse 1 / zwischen den Klassen). Die Spalte mit der Bezeichnung „Extremwerte“ enthält die Differenz zwischen dem maximalen Hamming-Abstand innerhalb der Klasse 1 und dem minimalen Hamming-Abstand zwischen den Klassen.

Bei den ersten beiden Belegungen der Parameter ist die Distanz zwischen dem maximalen Hamming-Abstand innerhalb der Klasse 1 und dem minimalen Hamming-Abstand zwischen den Klassen groß genug, um eine zuverlässige Unterscheidung der Klassen zu ermöglichen. Bei der dritten Belegung ist diese Distanz nicht ausreichend: der minimale Hamming-Abstand zwischen den Klassen ist kleiner als der maximale Hammin-Abstand innerhalb der
Klasse 1.

Da im Vergleich zum zweiten Parametersatz sowohl der Abstand der Mittelwerte, als auch der Abstand der Extremwerte für die erste Parameterbelegung signifikant besser ist, wurde für das weitere Vorgehen eine Anfangsbelegung von $\omega = 0.03125$ (bei $\alpha = 128$) festgelegt.

3. Untersuchung des Verhältnisses ω zu α

Wie in Abschnitt 4.4.3 dargelegt, wurde sich für die Untersuchung zweier verschiedener Verhältnisse von ω zu α entschieden. Daraus ergaben sich folgende Anfangsbelegungen für die Waveletparameter:

1. $\omega =$ 0.03125 , $\alpha =$ 128 , $\beta =$ 20
2. $\omega =$ 0.046875 , $\alpha =$ 128 , $\beta =$ 20

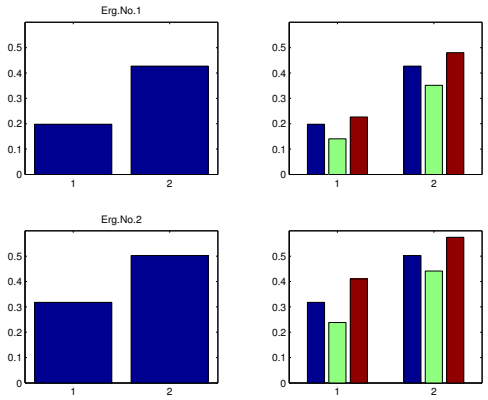


Abbildung 4.13: Ergebnis der Untersuchung des Verhältnisses ω zu α

Daraus ergaben sich folgende Differenzwerte für die mittleren Hamming-Abstände (siehe auch Abb. 4.13 und Anhang E.3):

	Mittelwerte	Extremwerte
1.	0,229166	0,12500
2.	0,184636	0,03125

Sowohl die Differenz der mittleren Hamming-Abstände, als auch die Differenz der Extremwerte ist für den ersten Parametersatz größer. Somit ist eine Verwendung des ersten Parametersatzes zum Zwecke des Vergleichs der Testdaten besser geeignet. Besonders der Unterschied zwischen dem maximalen Hamming-Abstand innerhalb der Klasse 1 zu dem minimalen Hamming-Abstand zwischen den Klassen ist im Falle des zweiten Parametersatzes als nicht ausreichend zu betrachten.

Somit hat sich die Verwendung eines Verhältnisses $\omega \cdot \alpha = 4$ experimentell bestätigt[17].

4. Untersuchung des Verhältnisse α zu β

Als letztes wurde der Einfluss der vertikalen Ausbreitung der Wavelets untersucht. Diese wird durch den Parameter β bestimmt:

1.	$\omega =$	0.03125	,	$\alpha =$	128	,	$\beta =$	30
2.	$\omega =$	0.03125	,	$\alpha =$	128	,	$\beta =$	20
3.	$\omega =$	0.03125	,	$\alpha =$	128	,	$\beta =$	10

Dabei kam es zu folgenden Hamming-Abständen:

	Mittelwerte	Extremwerte
1.	0,220833	0,12500
2.	0,229166	0,12500
3.	0,233073	0,15912

Es kann aus den Ergebnissen abgelesen werden, dass bei einer Verringerung der vertikalen Ausdehnung der Wavelets (was zu einer klareren Abgrenzung der vertikalen Bereiche der untersuchten Daten führt), die Distanzen der betrachteten Hamming-Abstände zunehmen. Somit erweist sich der dritte Parametersatz für eine Anwendung auf die Testdaten als am besten geeignet.

[17] Wie schon erwähnt sind solche Aussagen in den Begrenzungen der hier vorgenommenen Versuche zu betrachten.

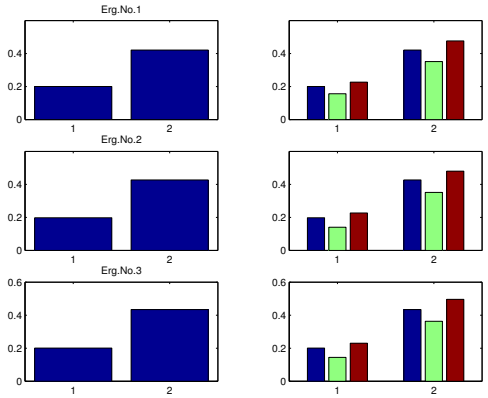


Abbildung 4.14: Ergebnis der Untersuchung des Verhältnisses α zu β

Es sollte aber beachtet werden, dass der mittlere Hamming-Abstand innerhalb der Klasse 1 für den dritten Parametersatz mit 0,200521 über dem mittlerem Hamming-Abstand für die zweite Belegung liegt, der 0,197917 beträgt (siehe die Einzelergebnisse in Anhang E.4). Dieses deutet darauf hin, dass sich die Zuordnung der Bilder zu der Referenziris, trotz einer verbesserten Unterscheidung zwischen den Klassen, verschlechtert hat.

Kapitel 5

Auswertung

Diese Kapitel enthält eine Auswertung der Resultate der theoretischen Betrachtungen aus Kapitel 3 und der praktischen Versuche aus Kapitel 4.

5.1 Auswertung der Versuche

Im folgenden sollen die Verfahren und die Ergebnisse aus Kapitel 4 betrachtet und analysiert werden.

5.1.1 Resultate der Vorüberlegungen

Wie schon in Kapitel 4 erläutert, besitzen Verfahren, die Wavelets einsetzen, eine große Anzahl von Freiheitsgrade, deren Abhängigkeit untereinander nicht klar ist. Somit ist eine eingehende Untersuchung der vorliegenden spezifischen Signaleigenschaften unerlässlich, damit die Komplexität der Kalibrierung eines Systems nicht zu groß wird.

Anhand des etablierten Algorithmus von John Daugman wurde demonstriert, wie eine Identifikation und Einschränkung der Freiheitsgrade durchgeführt wird. Die in Abschnitt 4.4.3 vorgestellten Methoden sind auch auf den Fall anwendbar, dass ein Vorwissen, wie es in Abschnitt 4.4.2 entwickelt wurde, nicht vorliegt.

Neben der Entwicklung dieses Vorgehens konnten, als Resultat der vorgenommenen Untersuchungen, weitere Erkenntnisse gewonnen werden, die sowohl ein tieferes Verständnis des Algorithmus, wie auch Einblicke in algorithmus-unabhängige Eigenschaften der Signalstruktur (d. h. der menschlichen Iris) ermöglichten:

Anpassung der Signalstruktur in der Vorverarbeitung

Die Transformation der Irisdaten in das pseudo-polare Koordinatensystem führt zu einer Konzentration der in den Bilddaten auftretenden Frequenzen. Treten vor der Transformation noch zu gleichen Anteilen Frequenzen aller Richtungen auf, entsteht nach der Transformation eine Konzentration auf die horizontalen Frequenzen.

Dies ist ein gutes Beispiel dafür, wie die Struktur der zu untersuchenden Daten vor einer Verarbeitung angepasst werden kann, um eine effiziente Untersuchung zu erzielen: Durch die Konzentration auf horizontale Daten kann auf eine Untersuchung der Daten mit Wavelets verschiedener Ausrichtung verzichtet werden; die Verwendung von horizontal ausgerichteten Wavelets ist ausreichend.

Dass die Autoren von [Zhu00a], deren Verfahren eine vergleichbare Vorverarbeitung verwendet (siehe 3.1.2), die Irisdaten mit Gaborwavelets verschiedener Ausrichtung untersuchen, lässt in diesem Kontext Zweifel an dem ebendort vorgestellten Verfahren aufkommen.

Struktur der Iris

Nicht direkt aus der vorgenommenen Untersuchung der Freiheitsgrade, sondern aus der vorausgehenden Teilimplementierung des Algorithmus entstammt die Beobachtung, dass die Pupille des menschlichen Auges nicht zentrisch in der Iris positioniert ist. Diese Information war in der Theorie zwar schon vor den Versuchen bekannt, die Versuche halfen jedoch die Auswirkungen dieses Umstands zu verstehen. Lässt man die zum Irismittelpunkt verschobene Position der Pupille unberücksichtigt, kommt es bei der Transformation der Daten zu Verwischungen der Irismuster (siehe Abb. 5.1).

Diese Erfahrung unterstützt die Erkenntnis, dass der Einfluss der Vorverarbeitung der Irisdaten auf die Güte des Algorithmus mindestens ebenso wichtig ist, wie die Parametrisierung.

Zusätzliche Freiheitsgrade durch die Vorverarbeitung

Die letzten beiden Punkte und die Bemerkung 4.5 zeigen in ihrer Kombination, dass neben den Freiheitsgraden, die aus den Wavelet-Eigenschaften resultieren, der Algorithmus weitere Variablen durch die Vorverarbeitung erhält. Diese Variablen erfordern ebenfalls ein hohes Maß an Vorüberlegung.

Abbildung 5.1: Verwischte Irismuster durch fehlerhafte Vorverarbeitung

5.1.2 Interpretation der Versuchsergebnisse

Die Versuche, die in Abschnitt 4.6 vorgenommen wurden, endeten mit dem Ergebnis, dass eine Parametrisierung mit den Anfangswerten $\omega = 0.031250$, $\alpha = 128$ und $\beta = 10$ zu der besten Leistung des Algorithmus auf den Testdaten führt. Eine Zusammenfassung der aufgetretenen Hamming-Abstände gibt folgende Tabelle an:

	Mittel	Min	Max
Innerhalb Klasse 1	0.200521	0.144531	0.230469
Zwischen Klasse 1/2	0.433594	0.363281	0.496094

Die Spalten der Tabelle mit den Bezeichnungen „Mittel“, „Min“ und „Max“ geben jeweils den mittleren, minimalen und maximalen Hamming-Abstand, für die Vergleiche innerhalb der Klasse der Referenziris („Innerhalb Klasse 1“) und zwischen den beiden Klassen („Zwischen Klasse 1/2“), an.

Vergleich mit den Ergebnissen Daugmans

Ein Vergleich dieser Versuchsergebnisse mit den, in Abbildung 5.2 angegebenen, Ergebnissen von John Daugman zeigt, dass die erreichten Hamming-Abstände sich annähernd innerhalb der Bereiche befinden, die Daugman ermittelt hat: Die Hamming-Abstände der Datensätze innerhalb der Klasse 1 befinden sich vollständig in dem von Daugman dokumentierten Bereich. Im Falle der Hamming-Abstände zwischen den Klassen, befinden sich der

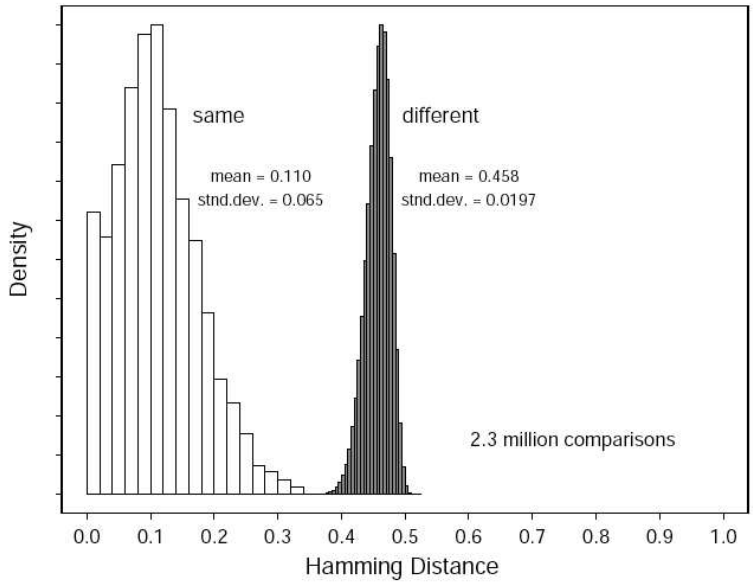


Abbildung 5.2: Ergebnis der Versuche von John Daugman [Daug93]

Durchschnittswert und der maximale Hamming-Abstand in ähnlichen Bereichen wie die Werte Daugmans. Der minimale Hamming-Abstand liegt unterhalb der von Daugman dokumentierten Grenze.

Diese Beobachtung lässt den Schluss zu, dass die Implementierung und Parametrisierung (innerhalb der angegebenen Grenzen) erfolgreich war. Wie in Kapitel 4 schon argumentiert, ist eine zuverlässige Validierung dieses Schlusses aufgrund der eingeschränkten Datenbasis nicht möglich. Die weiteren Betrachtungen sind unter diesen Einschränkungen zu sehen.

Folgerungen aus den Versuchsergebnissen

Bezogen auf die Testdaten lässt sich aus den Ergebnissen folgern, dass für eine Unterscheidung von Irisbildern Wavelets geeignet sind, deren horizontale Ausbreitung (bezüglich der transformierten Irisdaten) einen großen Bereich der untersuchten Daten abdeckt. Wie den Einzelergebnissen in Anhang E.4 entnommen werden kann, verfügen bei einer nach Waveletparameter aufgeteilten Betrachtung Wavelets mit einer Parametrisierung von $\alpha = 64$[1] über die beste Unterscheidung der Bildklassen.

[1] Dieses bezieht sich auf die Versuchsreihe mit den Anfangswerten $\omega = 0,03125$, $\alpha = 128$ und $\beta = 10$. Somit ist die vollständige Parametrisierung der angegebenen Wavelets mit $\omega = 0,0625$,$\alpha = 64$ und $\beta = 5$ gegeben.

Aus diesen Werten folgt, dass die horizontale Ausdehnung des Wavelets (siehe hierzu Abschnitt 4.6.2) etwa die Hälfte des Irisrings abdeckt. Die Auswirkung der vertikalen Ausdehnung ist weniger entscheidend. Bei den Versuchen mit verschiedenen Parametrisierungen der entsprechenden Variable β wiesen alle Varianten ausreichend gute Eigenschaften auf, mit einer Tendenz zur Verbesserung der Ergebnisse bei der Verwendung kleiner Werte.

Eine direkte Interpretation dieser Beobachtungen fällt schwer, da der Algorithmus lediglich die Phase der komplexen Waveletkoeffizienten zur Generierung der Signatur verwendet. Eine Einbeziehung der Resultate anderer Quellen erscheint in diesem Zusammenhang vielversprechend:

In [Wisk99] wird beschrieben, dass die Ausprägung der Phase der Waveletkoeffizienten stark von der Exaktheit ihrer Positionierung abhängt (siehe Abbschnitt 3.3.1). Die Autoren berichten, dass zueinander leicht verschobene „Jets“ signifikant unterschiedliche Phaseninformationen tragen können, obwohl sie fast identische Bilddaten beschreiben. Eine Übertragung dieses Umstandes auf die vorliegenden Ergebnisse lässt folgende Schlüsse zu: Je kleiner die Wavelets sind, desto stärker besteht eine Abhängigkeit von einer genauen Positionierung, da effektiv weniger Bildpunkte in die Berechnung des Waveletkoeffizienten einfließen. Aus den Aussagen in [Wisk99] folgt, dass sich eine schlechte Positionierung stark auf die Phase der Koeffizienten auswirkt (stärker als auf die Amplitude). Da der implementierte Algorithmus lediglich die Phaseninformationen berücksichtigt, kann die bessere Performance bei großen Wavelets auf die größere Toleranz dieser gegenüber einer Abweichungen der Position interpretiert werden. Dies deutet darauf hin, dass durch eine präzisere Vorverarbeitung, die in einer genaueren Positionierung der Analysepunkte resultiert[2], die Fähigkeit kleinerer Wavelets zunimmt, Bilddaten miteinander zu identifizieren.

Bemerkung 5.1 (Fortschritte Daugmans)
Abbildung 5.2 spiegelt die Ergebnisse Daugmans zum Zeitpunkt der Veröffentlichung dieser Grafik wieder. Wie [Daug99] entnommen werden kann, konnte eine Verbesserung der Leistung des Algorithmus Daugmans verzeichnet werden. Da keiner der Artikel Daugmans, die im Anschluss an [Daug93] veröffentlicht wurden, Veränderungen des Algorithmus enthielten, kann gefolgert werden, dass diese Verbesserung vollständig durch Änderungen der Parametrisierung erzielt wurden. Dieses belegt die Behauptung, dass eine gute Parametrisierung ein langwieriger Prozess ist (der sich im Falle Daugman über Jahre erstreckt).

[2] Eine genauere Positionierung versteht sich in diesem Fall bezüglich verschiedener Aufnahmen derselben Iris.

5.2 Theoretische Betrachtungen

Unterstützt von den eben formulierten Erkenntnissen aus der praktischen Arbeit, können anhand der in Kapitel 3 beschriebenen Algorithmen Untersuchungen vorgenommen werden, inwieweit Parallelen zwischen den verschiedenen Anwendungen von Wavelets in der biometrischen Authentikation existieren. In diese Betrachtungen fließen sowohl Erkenntnisse aus den fünf detailliert beschriebenen als auch aus den weiteren, nur kurz angesprochene Verfahren ein. Die in Abschnitt 5.2.4 vorgenommene Klassifikation der Algorithmen beschränkt sich somit auch auf die angegebene Menge von Algorithmen. Es wird keinesfalls ausgeschlossen, dass weitere waveletbasierte Algorithmen zur biometrischen Authentikation existieren, die sich nicht in diese Klassifikation eingliedern lassen.

5.2.1 Vergleich von Waveletkoeffizienten

Aus dem Überblick der Algorithmen, den Kapitel 3 vermittelt, ergibt sich zuerst die algorithmen-übergreifende Erkenntnis, dass alle betrachteten Verfahren darauf basieren, die Waveletkoeffizienten der vorgenommenen Wavelettransformationen zu vergleichen. Dabei unterscheiden sich Ausprägung und Verfahren des Vergleichs. So generieren beispielsweise die Verfahren von Daugman [Daug93] und Hamamoto [Hama99] die Signaturen direkt aus den Waveletkoeffizienten, während die Verfahren von Garcia et al. [Garc00] und Zhu et al. [Zhu00a] jeweils eine Reihe von Waveletkoeffizienten für ein Element der Signatur zusammenfassen.

5.2.2 Syntaktische Analyse

Die betrachteten Verfahren unternehmen in der Regel keine semantische Analyse der Waveletkoeffizienten (die Ausnahme bildet das Verfahren von Garcia et al.; mehr dazu im folgenden). Die Verwendung der Koeffizienten beinhaltet kein „Verstehen“ der Bildstrukturen. Es findet keine direkte Zuordnung von Waveletkoeffizienten zu abstrakten Elementen des untersuchten Merkmals statt (was z. B. einer Interpretation eines gegebenen Waveletkoeffizienten bzw. der Zusammenfassung einer Reihe von Koeffizienten als Minutie eines Fingerabdrucks entspräche). Das Verfahren von Wiskott et al. [Wisk99] bildet hier keine Ausnahme. Zwar findet dort eine Zuordnung von „Jets“ zu Elementen des Gesichts (Mund, Nase, etc.) statt, doch die Behandlung der „Jets“ ist rein abstrakt und die initiale Zuweisung eines Jets zu einem Gesichtselement geschieht nicht über eine Interpretation der Koeffizienten, sondern über die bewusst gesteuerte Positionierung des Jets während der Parametrisierung des Algorithmus.

Wie schon angedeutet, stellt das Verfahren von Garcia et al. zur Gesichtserkennung in Einschränkungen eine Ausnahme dieser Beobachtung dar. Das Verfahren verwendet die Waveletkoeffizienten für zwei separate Aufgaben: Die Lokalisierung der Gesichtsregionen und die Generierung der Signatur. Im Falle der Generierung der Signatur trifft die oben beschriebene Beobachtung zu: eine semantische Analyse der Koeffizienten findet nicht statt. Anders verhält sich jedoch der Vorgang der Lokalisierung der Regionen des Gesichts. Hierfür werden Erkenntnisse über charakteristische Muster von Waveletkoeffizienten verwendet. So geschieht z. B. das Finden der Begrenzungen des Gesichts über eine Lokalisierung von Maxima bestimmter Waveletkoeffizienten.

5.2.3 Vergleichsverfahren

Auch wenn alle betrachteten Verfahren algorithmisch auf einem Vergleich der Waveletkoeffizienten beruhen, unterscheiden sie sich doch in der Art des Vergleichs dieser Koeffizienten. Die Bandbreite der Methoden reicht vom reinen Bitvergleich (bei Daugman), über statistische Methoden (wie z. B. bei Zhu et al. und Garcia et al.) bis zu der Verwendung von neuronalen Netzen (beim Verfahren von [Lim01]).

5.2.4 Versuch einer Klassifikation

Algorithmen für die biometrische Authentikation, deren Methode des Signaturenvergleichs auf einem Vergleich von Waveletkoeffizienten basiert, lassen sich aufgrund der oben zusammengefassten Erkenntnisse in zwei Klassen unterteilen. Die Aufteilung geschieht auf Basis der verwendeten Technik der Signaturberechnung und der daraus folgenden Vergleichsmethode. Diese Klassifikation soll helfen, Parallelen zwischen Verfahren zu erkennen, die verschiedene biometrische Techniken realisieren.

1. Globale Verfahren

Alle unter dieser Überschrift zusammenfassten Algorithmen besitzen die Eigenschaft, dass sie für die von ihnen verwendete Merkmalscharakterisierung eine verteilte Menge von Waveletkoeffizienten nutzen, deren Ursprungsposition sich über das gesamte Merkmal (oder große Flächen) erstreckt. Die Algorithmen [Zhu00a], [Garc00] und [Bole98] gehören dieser Klasse an. Diesen Algorithmen ist gemein, dass sie die Waveletkoeffizienten bezüglich eines Mitgliedes der verwendeten Waveletfamilie zusammenfassen und das Resultat dieser Zusammenfassung als ein Element der biometrischen Signatur ablegen. Die Zusammenfassung der Koeffizienten erfolgt über statistische

Methoden wie der Mittelwertbildung oder der daraus abgeleiteten Berechnung der zugehörigen Standardabweichung.

Bei den Algorithmen [Liu01] und [Lim01] findet keine direkte Zusammenfassung der Koeffizienten einer Waveletfamilie statt. Die Vergleichsmethoden verwenden jedoch die Gesamtheit der Koeffizienten (in [Lim01]) bzw. alle Koeffizienten bezüglich eines Wavelets (in [Liu01]) für den Vergleich. Eine regionale Aufspaltung der Koeffizienten findet nicht statt.

2. Lokale Verfahren

Die Verfahren von Hamamoto und Wiskott et al. gleichen sich in dem Ansatz, an einigen festgelegten Punkten eine Reihe von Koeffizienten bezüglich verschiedener Wavelets[3] zu berechnen. Über diese Waveletkoeffizienten wird eine Charakterisierung der Regionen erzeugt, deren Zentren diese Punkte sind. Eine mathematische Zusammenfassung der Koeffizienten wird nicht vorgenommen. Es findet somit ein direkter Vergleich von Waveletkoeffizienten statt, die sich auf dieselbe eingeschränkte Region der Daten beziehen. Dasselbe Verfahren verwendet auch Daugman, mit dem Unterschied, dass bei seiner Methode nicht bekannt ist, wie sich die Analysepunkte auf die verschiedenen Wavelets verteilen (siehe Abschnitt 4.4.3).

Diesen Verfahren ist gemein, dass die Güte ihrer Erkennung von der Güte der Positionierung der Wavelets abhängt. Besonders bei den Verfahren Daugmans und Hamamotos, die zur Festlegung der Analysepunkte auf Methoden basieren, die abgelöst von der Wavelettransformation sind, besteht somit eine starke Abhängigkeit von der Vorverarbeitung.

5.2.5 Unterschied zu „klassischen“ Verfahren

„Klassische“ Verfahren der Biometrik basieren auf einem Verständnis der Eigenschaften des untersuchten Merkmals. So werden bei der Fingerabdruckerkennung die einzelnen Muster erkannt und klassifiziert. Die Methoden der biometrischen Erkennung über die Handgeometrie setzten eine zuverlässige Zuordnung der aufgenommenen Daten zu den einzelnen Elementen der Hand voraus. Ein „klassisches“ Verfahren ist vollständig auf ein Merkmal festgelegt, da seine Funktionsweise auf die spezifischen Eigenschaften dieses Merkmals abgestimmt ist (die Methoden der Handgeometrie unterscheiden sich beispielsweise fundamental von den Methoden der Analyse der Retinastruktur).

Wie oben bereits erläutert, findet diese Art des „Verstehens“ der Daten im Allgemeinen bei waveletbasierten Methoden nicht statt. Dieses führt dazu,

[3] Das Attribut „verschieden“ bezieht sich hierbei nicht auf verschiedene Waveletfamilien, sondern auf einzelne Mitglieder derselben Familie, die sich in ihrer Parametrisierung (Größe, Ausrichtung, mittlere Frequenz) unterscheiden.

dass sich waveletbasierte Methoden z.T. merkmalsübergreifend übertragen lassen. Es können, wie im vorraus gegangenen Abschnitt beschrieben, Parallelen zwischen den behandelten Waveletverfahren gefunden werden: Das Verfahren Hamamotos zur Fingerabdruckerkennung ist eng mit der Methode Daugmans zur Iriserkennung verwandt und das Vergleichsverfahren von Zhu et al. für die Iriserkennung ähnelt der Methode für die Gesichtserkennung, die von Garcia et al. verwendet wird. Diese Eigenschaft ist zugleich die Stärke und der Schwachpunkt dieser Verfahren.

Die Stärke resultiert aus der Beschränkung der „intelligenten“ Analyse der Daten: Einer intelligenten Analyse muss eine vernunftgesteuerte Untersuchung und Klassifizierung der möglicherweise auftretenden Eigenschaften vorausgehen. Ist dieses bei einer eingeschränkten Menge an Ausprägungen, wie z. B. bei den Minutien des Fingerabdrucks, noch zu leisten, fällt eine derartige Untersuchung bei komplexeren Daten, wie den Irismustern, jedoch schwer. Bis dato wirde noch kein etablierter Algorithmus zur Iriserkennung eingeführt, der auf einer semantischen Analyse der Irismuster basiert.

Die Schwäche dieser Methode ist die eingeschränkte Beherrschbarkeit der Erkennung. Da kein wirkliches Verständnis der Strukturen vorhanden ist bzw. im Prozess der Bearbeitung und des Vergleichs kein Verstehen des Bildes vollzogen wird, lassen sich zuverlässige Aussagen über die Güte des Verfahrens nur schwer fällen. Sämtliche Angaben Daugmans über die Zuverlässigkeit seines Algorithmus entstammen statistischen Auswertungen seiner praktischen Versuchergebnisse.

Kapitel 6

Zusammenfassung & Ausblick

6.1 Zusammenfassung der Resultate

Die zu Beginn formulierte Zielsetzung dieser Arbeit war die Schaffung eines tiefgreifenden Verständnisses waveletbasierter Algorithmen zur biometrischen Authentikation. Zu diesem Zweck gab Kapitel 2 ein Einführung in die mathematischen Hintergründe der Wavelettheorie. In Kapitel 3 wurden fünf ausgewählte Algorithmen vorgestellt und auf ihre Funktionsweise untersucht. In Kapitel 4 wurde die Implementierung und Parametrisierung eines dieser Algorithmen vorgestellt, um die praktischen Aspekte waveletbasierter Algorithmen zu beleuchten. In Kapitel 5 wurden die erzielten Ergebnisse ausgewertet und in Relation zueinander gesetzt.

Eine Zusammenstellung der Erkenntnisse über das Feld der Anwendungen von Wavelets in der biometrischen Authentikation gestaltet sich wie folgt:

- Breite Anwendbarkeit

  Für viele Bereiche der biometrischen Authentikation wurden waveletbasierte Algorithmen entworfen. Die Wavelets werden dabei sowohl zur Signaturerzeugung wie auch zur Vorverarbeitung der Daten angewendet.

- Mathematisches Fundament

  Die Verwendung der Wavelets steht auf einem festen mathematischen Fundament. Richtig angewendet erlauben die Wavelets eine effiziente, verlustfreie Transformation der Daten.

- Entwicklungsfähigkeit

  Der Großteil der betrachteten Algorithmen verwendet nur wenige der Methoden der fortgeschrittenen Wavelettheorie oder setzt die benutzen Wavelets in Bezug zu dieser. Dies deutet darauf hin, dass die Entwicklung der waveletbasierten Algorithmen zur biometrischen Authentikation noch nicht ausgeschöpft ist.

- Dominanz der syntaktischen Analyse

  Wie in Kapitel 5 ausgeführt, werden im Großteil der betrachteten Algorithmen die Waveletkoeffizienten lediglich auf einem abstrakten Niveau ausgewertet. Eine semantische Relation von Waveletkoeffizienten zu tatsächlichen Eigenschaften des Merkmals findet meistens nicht statt.

- Große Anzahl von Freiheitsgraden

  Die in Kapitel 3 vorgenommene Analyse der Algorithmen ergab, dass neben dem eigentlichen algorithmischen Verfahren ein ebenso großes Gewicht auf der erfolgreichen Parametrisierung der Algorithmen liegt. Dieses liegt an der großen Anzahl von Freiheitsgraden, die waveletbasierten Methoden eigen ist.

- Methoden der Parametrisierung

  Eine Konsequenz aus der großen Anzahl der Freiheitsgrade ist eine grundlegende Bedeutung der Methoden zur Bestimmung des Wertebereichs dieser Freiheitsgrade. In Kapitel 4 wurde ein systematisches Vorgehen zu diesem Zweck vorgestellt. Ein besonderes Gewicht liegt hierbei auf einer genauen Untersuchung der spezifischen Eigenschaften der zu untersuchenden Daten (in Bezug auf das Anwendungsziel).

- Bedeutung der Vorverarbeitung

  Die Untersuchungen der Verfahren von Daugman [Daug93], Zhu et al. [Zhu00a] und Hamamoto [Hama99] haben eine, für die Güte des Algorithmus entscheidende, Bedeutung der Vorverarbeitung der Daten ergeben. Gerade bei lokal ausgerichteten Verfahren, wie der Methoden von Daugman und Hamamoto, ist eine exakte Positionierung der Analysepunkte essenziell. Die praktischen Versuche in Kapitel 4 haben dieses belegt.

- Ermittlung von Parallelen

  Durch die theoretischen Betrachtungen in Kapitel 3 konnten algorithmusübergreifende Parallelen zwischen den betrachteten Verfahren festgestellt werde. Dieses führte zu allgemeineren Schlüssen über die Funktionsweise waveletbasierter Algorithmen zur biometrischen Authentikation. Dieses abstrakte Wissen kann sowohl für das Verständnis neu-

er Algorithmen, als auch bei der Verbesserung bestehender Methoden hilfreich sein.

Eine abschließende Beurteilung der Klasse der waveletbasierten Algorithmen kann an dieser Stelle nicht gegeben werden. Für eine derartige Aussage ist dieses Feld zu komplex und zu jung. An vielen Stellen entsteht der Eindruck, dass die Entwicklung waveletbasierter Verfahren im Gebiet der Biometrik noch am Anfang steht. Die etablierten Verfahren verwenden nur wenige Erkenntnisse und Methoden der Wavelettheorie. Eine mögliche Richtung der Entwicklung zeigt das Verfahren von Garcia et al. auf, das in Abschnitt 3.3.2 beschrieben wurde: Das Verfahren verwendet fortgeschrittene Methoden der Wavelettheorie (die „Wavelet Packets“) und nimmt eine Interpretation der Waveletkoeffizienten zur Bestimmung definierter Regionen des Gesichtes vor. Ob eine derartige Entwicklung zu einer weiteren Verbesserung der Verfahren und einer besseren Beherrschbarkeit der Methoden führt, ist zu vermuten, kann aber nicht mit Sicherheit prognostiziert werden.

6.2 Ausblick

Ein Ausblick dieser Arbeit gestaltet sich zweigeteilt. Der erste Teil des Ausblicks behandelt Aspekte, die sich direkt aus dem Inhalt dieser Arbeit ergeben. Der zweite Teil stellt das betrachtete Feld in einen breiteren Kontext.

6.2.1 Fortführung der praktischen Arbeit

Die in Kapitel 4 vorgestellte Arbeit bieteT verschiedene Möglichkeiten der Weiterführung:

- Validierung durch größere Sampleanzahl

  Sämtliche Ergebnisse, die im Rahmen dieser Arbeit erzielt wurden, sind unter der Einschränkung zu sehen, dass die Anzahl der verwendeten Irisbilder sehr begrenzt war. Wie schon erwähnt, diente der praktische Teil dieser Arbeit mehr der Konzeption und Bereitstellung von Methoden und Werkzeugen für eine Parametrisierung als der Durchführung der eigentlichen Auffindung der Parameter. Eine Weiterführung der Experimente mit einer erhöhten Anzahl von Irissamples wäre wünschenswert, um so einer Referenzimplementierung des Algorithmus näher zu kommen.

- Untersuchung des Informationsgehalts verschiedener Regionen der Iris

  Aus Aufwandsgründen hat diese Arbeit auf genauere Untersuchungen verzichtet, die die Verteilung der Analysepunkte bezüglich der Irisdaten betreffen. Dieser Bereich bietet interessante Aspekte zur weiteren

Erforschung. Dieses kann sowohl empirisch (statistische Verteilung von Irismuster auf den Regionen der Iris), informationstheoretisch (Betrachtungen zu der Entropie der Irisregionen u. ä.) und experimentell (Versuchreihen mit den im Rahmen dieser Arbeit entwickelten Werkzeugen) geschehen.

- Anwendung fortgeschrittener Erkenntnisse der Wavelettheorie

  In Abschnitt 4.4.3 wurde hinsichtlich der Zuordnung von Wavelets zu Regionen der Iris angeregt, Methoden zu verwenden, die an die diskrete Wavelettransformation angelehnt sind. Dieses bezieht sich insbesondere auf die Untersuchung von wenigen Punkten mit „großen“ Wavelets und von vielen Punkten mit „kleinen“ .

- Erstellung von Vorverarbeitungsroutinen

  Ein Punkt, der die gestellten Aufgaben lediglich am Rand berührte, ist die Erstellung der Routinen zur Vorverarbeitung der Irisdaten. Wie jedoch bereits dargelegt wurde, haben diese Routinen für die Güte einer Implementierung mindestens genauso viel Bedeutung, wie der eigentliche waveletbasierte Algorithmus. Eine vollständige Implementierung dieser Routinen erscheint als interessante Aufgabenstellung.

Weiterführende praktische Untersuchungen

Voraus gesetzt, dass eine Implementierung und Parametrisierung erfolgreich war und eine Referenzimplementierung des Algorithmus von Daugman vorliegt, können weiterführende praktische Versuche unternommen werden. Diese sollten auf ein tieferes Verständnis des Algorithmus und eine eventuelle Weiterentwicklung abzielen:

- Veränderung der Konstanten des Verfahrens

  Verbessert sich die Güte der Erkennung, wenn die Anzahl der untersuchten Punkte (z. B. über das Einfügen weiterer Analyse-Bänder) erhöht wird?

  Wie weit verschlechtert sich die Qualität der Erkennung, wenn die Anzahl der untersuchten Punkte verringert wird? In welchem Verhältnis steht der Gewinn an Performance zu dem zu erwartenden Verlust an Erkennungsgenauigkeit?

- Andere Wavelets

  Welchen Einfluss hat die Verwendung anderer Wavelets im Algorithmus? Ist die Funktionsfähigkeit des Verfahrens vollständig auf Gaborwavelets festgelegt oder kann auch mit anderen Wavelets ein ähnliches

Verhalten erzielt werden? Von besonderem Interesse ist bei einer derartigen Aufgabestellung die anschließende Rückkopplung mit den theoretischen Erkenntnissen der Wavelettheorie.

- Andere Vergleichsmethoden

  Welches Gewicht besitzt die von Daugman verwendete Vergleichsmethode in der Erkennung? Welche Erkennungsgenauigkeit kann mit einer Methode erzielt werden, die statt der Phase z. B. die Amplitude der Waveletkoeffizienten verwendet?

Sollte eine Referenzimplementierung des Algorithmus zur Verfügung stehen, böte sich außerdem an, Versuche zu unternehmen, welche die Robustheit des Algorithmus gegenüber Täuschungsversuchen evaluieren.

6.2.2 Weiterführende Aspekte von Anwendungen der Wavelets in der biometrischen Authentikation

Losgelöst von der praktischen Seite dieser Arbeit, stellen sich eine Reihe von weitergehenden Ansätzen und Fragestellungen für eine Weiterbearbeitung dieses Themas:

- Untersuchung anderer biometrischer Techniken

  Die Betrachtungen in Kapitel 3 blieben auf die Felder der Iris-, Fingerabdruck- und Gesichtserkennung beschränkt. Von Interesse ist eine Untersuchung, inwieweit waveletbasierte Methoden für andere biometrische Gebiete existieren. Neben anderen Anwendungen, die auf der Verarbeitung von Bildinformationen basieren, bietet sich besonders die Untersuchung von Wavelets für die Stimmerkennung an (Die Arbeiten D. Gabors, auf den die Gaborwavelets zurück gehen, beschäftigen sich mit der Analyse von Klängen).

- Methoden der „intelligenten“ Analyse

  Wie schon zuvor argumentiert, besitzt eine Interpretation der Waveletkoeffizienten hinsichtlich der Daten, die von ihnen repräsentiert werden, das Potential einer Aufwertung der biometrischen Algorithmen.

- Verwendung fortgeschrittener Waveletmethoden

  Ebenso interessant ist eine Untersuchung der Anwendbarkeit fortgeschrittener Wavelettechniken im Gebiet der biometrischen Authentikation. Der Algorithmus von Garcia et al. [Garc00] ist ein Schritt in diese Richtung.

Zusammenfassend lässt sich sagen, dass die Anwendung von Wavelets in der biometrischen Authentikation interessante Ansätze enthält und vielversprechende Verfahren ermöglicht. Wie soeben gezeigt, bieten sich diverse Ansatzpunkte für weitere Arbeiten. In diesem jungen Forschungsfeld sind auf unabsehbare Zeit Fortschritte zu erwarten.

Anhang A

Mathematischen Notation

A.1 Anpassung der Schreibweise

Im Laufe dieser Arbeit wurde aus verschiedenen Quellen mathematische Inhalte zitiert. Da die Notationen und Bezeichnungen in den Quellen uneinheitlich sind, war es für die Verständlichkeit und Lesbarkeit dieser Arbeit nötig, in einigen Fällen die Schreibweise der zitierten Formeln anzupassen und zu vereinheitlichen.

Ein Beispiel für unterschiedliche Notationen mathematischer Inhalte, ist die Verwendung der Exponentialfunktion: In einigen Quellen (z. B. [Hama99]) wird diese durch $\exp[x]$, in anderen (wie z.B. [Mall98]) durch e^x dargestellt. Ebenso verwenden einige Autoren die Vektorschreibweise zur Darstellung zweidimensionaler Datenstrukturen (wie z.B. Bilddaten), andere verzichten auf diese Art der Darstellung.

Es wurde darauf geachtet, dass lediglich die Schreibweisen, aber in keinem Fall die Aussagen der Formeln geändert wurde.

Im folgenden ist eine kleine Aufstellung der häufigsten Bezeichnungen in dieser Arbeit gegeben:

- $I(x, y)$: Bilddaten
- $\psi()$: Wavelet
- ϑ: Winkel
- ξ, ω: Frequenz

Falls diese Bezeichner in anderer Bedeutung Verwendung finden, wird dieses explizit angegeben oder es ist aus dem Zusammenhang selbsterklärend.

Eine umfassende Auflistung der verwendeten mathematischen Notationen folgt im nächsten Abschnitt.

A.2 Liste verwendeter mathematischer Symbole

$\mathbb{N}$	die natürlichen Zahlen
$\mathbb{Z}$	die ganzen Zahlen
$\mathbb{R}$	die rellen Zahlen
$\mathbb{C}$	die komplexen Zahlen
i	die imaginäre Zahl i
$V_1 \oplus V_2$	direktes Produkt der Vektorräume V_1 und V_2: $V_1 \oplus V_2 := \{v \mid v = v_1 + v_2 \ , \ v_1 \in V_1 \ , \ v_2 \in V_2\}$
$Span((v_i)_{i \in I})$	Der von der Menge v_i aufgespannte K-Vektorraum: $Span((v_i)_{i \in I}) := \{v \mid v = \sum_{i \in I} \lambda_i v_i \ , \ \lambda_i \in K\}$
$\bigcup$	Vereinigung
$\bigcap$	Schnittmenge / logisches Und (AND)
$\otimes$	Tensorprodukt / Logisches exklusives Oder (XOR)
$\in$	Element
$\subset$	Obermenge
$\supset$	Untermenge
$\int$	Integral
$\sum$	Summe
$f(t)$	Funktion/Signal
$\hat{f}(\omega)$	Fouriertransformierte von $f(t)$
$Wf(t)$	Wavelettransformierte von $f(t)$
$f \star g(u)$	Faltung (siehe Anhang B.2)
$\forall$	Für alle [Elemente]
$\exists$	Es existiert ein [Element]
v^T	Transponierte des Vektors v
z^*	Komplex-konjugierte von $z \in \mathbb{C}$
$\delta(t)$	Diracsche Deltafunktion oder Dirac-Stoß: $\int\limits_{-\infty}^{\infty} f(t)\delta(t) := f(0)$

Anhang B

Weitere mathematische Grundlagen

In diesem, die mathematischen Grundlagen ergänzenden, Abschnitt folgen Definitionen verschiedener mathematischer Begriffe, die keinem der anderen Abschnitte zugeordnet werden konnten, aber für das Verständnis der Arbeit nötig sind.

B.1 Lineare Algebra

Als Quelle für die folgenden Abschnitte diente, soweit nicht anders angegeben [Fisc79].

B.1.1 Vektorräume

Definition B.1 (Vektorraum) *Sei K ein Körper. Ein K-Vektorraum besteht aus einer Menge V, einer Verknüpfung + (Addition) und einer Verknüpfung $\cdot$ (Multiplikation mit Skalaren), so dass folgendes gilt:*

(V1) (V,+) ist eine Abelsche Gruppe

(V2) für alle v, $w \in V$, λ, $\mu \in K$ gilt

a. $(\lambda + \mu) \cdot v = (\lambda \cdot v) + (\mu \cdot v)$

b. $\lambda \cdot (v + w) = \lambda v + \lambda w$

c. $(\lambda\mu) \cdot v = \lambda(\mu \cdot v)$

d. $1 \cdot v = v$

Die im Rahmen dieser Arbeit auftretenden Vektorräume sind Funktionsräume[1] über $\mathbb{R}$ und $\mathbb{C}$. Es handelt sich dabei also um jeweils $\mathbb{R}$-Vektorräume (reeller Vektorraum) oder $\mathbb{C}$-Vektorräume (komplexer Vektorraum).

Ein reeller Vektorraum V für den ein Skalarprodukt

$$\begin{array}{rcl} <,>: V \times V & \rightarrow & \mathbb{R} \\ (v,w) & \mapsto & <v,w> \end{array}$$

erklärt ist (für eine genaue Definition des Skalarprodukts siehe [Fisc79, S. 184]), nennt sich *euklidischer* Vektorraum. Ein komplexer Vektorraum mit einem Skalarprodukt heißt *Unitär*. Zwei Elemente eines Vektorraums heißen *orthogonal* zueinander, wenn ihr Skalarprodukt gleich Null ist.

In dem in dieser Arbeit behandelten Fall der Funktionsräume wird folgendes Skalarprodukt verwendet:

$$<f,g>= \int_{-\infty}^{+\infty} f(x)g(x)\, dx. \tag{B.1}$$

Für jeden Vektorraum existiert eine Menge von Elementen, die diesen aufspannen:

Definition B.2 (Erzeugendensystem, Basis) *Gegeben sei ein K-Vektorraum V mit einer Familie $(v_i)_{i\in I}$ von Vektoren. Die Familie heißt Erzeugendensystem von V, falls gilt*

(B1) $V = Span((v_i)_{i\in I})$.

Ein Erzeugendensystem heißt Basis, falls gilt

(B2) $(v_i)_{i\in I}$ ist linear unabhängig.

Eine Basis heißt *Orthogonalbasis*, wenn alle Elemente der Basis orthogonal zueinander sind. Eine Basis heißt *Orthonormalbasis*, wenn alle Elemente der Basis eine Norm von Eins haben und orthogonal zueinander sind.

Im Falle nicht orthogonaler Basen von nicht endlicher Dimensionalität ist es wichtig, auf die *Stabilität* dieser Basen zu achten. Diese Bedingung erfüllen die *Riesz-Basen* (siehe [Mall98, Appendix A.3]).

B.1.2 Norm und Metrik

Ein Maß für Elemente eines Vektorraums bildet die *Norm*:

[1] Das bedeutet, dass die Elemente dieser Räume reelle bzw. komplexe Funktionen sind.

Definition B.3 (Norm) *Sei V ein K-Vektorraum. Eine Abbildung*

$$\begin{array}{rcl} \|\ \| : V & \to & \mathbb{R} \\ v & \mapsto & \|v\| \end{array}$$

heißt Norm auf V, falls für alle $v, w \in V$ und $\lambda \in K$ gilt:

$$\begin{array}{lrcl} N1 & \|\lambda v\| & = & |\lambda| \cdot \|v\| \\ N2 & \|v + w\| & \leq & \|v\| + \|w\| \\ N3 & \|v\| & = & 0 \iff v = 0 \end{array}$$

Die reelle Zahl $\|v\|$ heißt Norm des Vektors v.

Der Begriff der *Metrik* definiert ein Abstandsmaß auf Elemente einer Menge:

Definition B.4 (Metrik) *Sei X eine Menge. Eine Abbildung*

$$\begin{array}{rcl} d : X \times X & \to & \mathbb{R} \\ (x, y) & \mapsto & d(x, y) \end{array}$$

heißt Metrik auf X, falls für alle $x, y, z \in X$ gilt:

$$\begin{array}{lrcl} M1 & d(x, y) & = & d(y, x) \\ M2 & d(x, z) & \leq & d(x, y) + d(y, z) \\ M3 & d(x, y) & = & 0 \iff x = y \end{array}$$

$d(x, y)$ heißt Abstand oder Distanz zwischen x und y.

Ist $\|\ \|$ eine Norm auf dem K-Vektorraum V, so ist durch

$$d(v, w) = \|v - w\| \text{ für } v, w \in V \tag{B.2}$$

eine Metrik auf V erklärt.

B.2 Faltung

Die Grundlagen der im folgenden kurz zusammengefassten Eigenschaften der Faltung finden sich u. a. in [Meyer98] und [Mall98]. Diese Werke bildeten die Quelle für diesen Abschnitt.

Definition B.5 (Faltungsintegral) *Das Faltungsintegral ist wie folgt definiert:*

$$f(t) = g \star h(t) = \int_{-\infty}^{\infty} g(u)h(t - u)\, du. \tag{B.3}$$

Eine wichtige Eigenschaft der Faltung ergibt sich im Zusammenspiel mit der Fouriertransformation:

$$f(t) = g \star h(t) \leftrightarrow \hat{f}(\omega) = \hat{g}(\omega)\hat{h}(\omega). \tag{B.4}$$

Die Faltung ist kommutativ:

$$g \star h(t) = h \star g(t). \tag{B.5}$$

Daraus folgt:

$$\int_{-\infty}^{+\infty} g(u)h(t-u) = \int_{-\infty}^{+\infty} h(u)g(t-u). \tag{B.6}$$

B.3 Statistische Methoden

In dem folgenden Abschnitt werden kurz mathematische Begriffe aus dem Bereich der Statistik definiert. Als Quelle für diesen Abschnitt diente [Lips74].

Einige der in Kapitel 3 vorgestellten Algorithmen verwenden den *Mittelwert* („Mean“) und die zugehörige *Standardabweichung* („Standard Deviation“) der Waveletkoeffizienten zur Generierung der Signaturen. Die zugrundeliegenden mathematischen Konzepte basieren auf dem Begriff der *Zufallverteilung* (oder auch Zufallsvariable).

Die allgemeingültige Definition des Begriffs der Zufallsverteilung ist für eingeschränkte Verwendung innerhalb dieser Arbeit zu weit gefasst. Im folgenden soll unter einer Zufallsverteilung X lediglich eine Funktion verstanden werden, die einem Element x einer diskreten Menge S eine Eintrittswahrscheinlichkeit $P(x)$ zuordnet.

Angewandt auf den Fall der Menge der Waveletkoeffizienten einer Transformation[2] bedeutet dieses, dass S die Menge der auftretenden Koeffizienten ist und $P(x)$ die Wahrscheinlichkeit angibt, dass bei einer zufälligen Auswahl eines Waveletkoeffizienten dieser gleich x ist.

Beispiel B.1 (Zufallverteilung)
Eine diskrete Transformation von sechs Analysepunkten hat als Ergebnis folgende Menge an Koeffizienten:

$$S = \{0,\ 0.134,\ 0,\ 0.23,\ 0.7,\ 0\}$$

[2] Die Menge der Waveletkoeffizienten einer durch ein Computerprogramm durchgeführten Wavelettransformation ist in jedem Fall diskret und endlich.

.

Daraus folgt eine Zufallverteilung X von

x	0	0.134	0.23	0.7
$P(x)$	$\frac{3}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$

Definition B.6 (Mittelwert) *Für eine gegebene Zufallsverteilung X für eine Menge S errechnet sich der Mittelwert $E(X)$ wie folgt:*

$$E(X) = \sum_{x_i \in S} x_i P(x_i). \tag{B.7}$$

Aus dem Mittelwert leitet sich die Standardabweichung ab:

Definition B.7 (Varianz, Standardabweichung) *Die Varianz $Var(X)$ einer Zufallsverteilung X für eine Menge S ist wie folgt definiert:*

$$Var(X) = \sum_{x_i \in S} (x_i - \mu)^2 f(x_i) = E((X - \mu)^2), \tag{B.8}$$

wobei $\mu = E(X)$ gilt. Die Standardabweichung σ_X ist gleich der Quadratwurzel der Varianz:

$$\sigma_X = \sqrt{Var(X)}. \tag{B.9}$$

Anhang C

Details der Implementierng

Dieser Teil des Anhangs beschreibt die angefertigte Implementierung auf einem quellcode-nahem Niveau. Die Beschreibung findet Routinenweise statt, wobei zur Strukturierung die in Abschnitt 4.2.3 eingeführte Modularisierung verwendet wird. Die Spezifikationen zur Implementierung aus Abschnitt 4.5.3 werden als bekannt vorausgesetzt. Im Vordergrund steht dabei die Beschreibung der Schnittstellen der Routinen. Alle hier beschriebenen Routinen sind „Matlab-Routinen“[1].

Der Quellcode der im folgenden beschriebenen Routinen kann im Internet unter `http://www.nerdalert.de/biometrik/daugman_matlab.tgz` erhalten werden.

C.1 P-Modul

C.1.1 loadIrisImage.m

Aufgabe

Laden eines Irisbildes und Umwandeln des Bildes in ein definiertes Format. In der momentanen Version liest die Routine Bilder im „bmp“-Format.

Übergabeparameter

Name	Typ	Inhalt
filename	String	Pfad zum zu ladenden Irisbild.

[1] Dieses gilt auch für die C++-Routine `mygabortransform.cpp`, da diese darauf ausgelegt ist, von Matlab aus aufgerufen zu werden. Nach einer Compilation mit dem Matlab-Compiler „mex“ steht diese Routine innerhalb von Matlab zur Verfügung.

Rückgabeparameter

Name	Typ	Inhalt
bild	[MxNx3]-Matrix	RGB Bild.

C.1.2 getGrayImage.m

Aufgabe

Umwandeln des übergebenen RGB-Bildes in ein Graustufenbild. Dabei kommt folgende Formel zur Berechnung der Graustufen zum Einsatz (Pixelweise):

Grauwert = 0.299· Rotwert + 0.587· Grünwert +0.114· Blauwert

Die Werte der Graustufen werden auf den Bereich $[0, 1]$ normiert.

Übergabeparameter

Name	Typ	Inhalt
bild	[MxNx3]-Matrix	RGB-Bild.

Rückgabeparameter

Name	Typ	Inhalt
grauBild	[MxN]-Matrix	Graustufenbild.

C.2 Q-Modul

C.2.1 getIrisBoundarys.m

Aufgabe

Anhand der Metadaten (siehe Abschnitt 4.5.3) des Irisbildes werden der Pupillenmittelpunkt und die Radien der Irisgrenzen errechnet. Dabei werden für die äußere Grenze (zwischen Iris und Sklera) der Radius bezüglich des linken und des rechten Irisrandes berechnet.

Übergabeparameter

Name	Typ	Inhalt
filename	String	Pfad des Files der Metadaten.

Rückgabeparameter

Name	Typ	Inhalt
pupillenMittelpunkt	Koordinate	Mittelpunkt der Pupille.
pupillenRadius	Fließkommawert	Radius der Grenze Pupille/Iris.
skleraRadius1	Fließkommawert	Radius der Grenze Iris/Sklera bezüglich des linken Irisrands.
skleraradius2	Fließkommawert	Radius der Grenze Iris/Sklera bezüglich des rechten Irisrands.

C.2.2 irisTransform.m

Aufgabe

Umwandlung der Irisdaten in eine rechtwinklige Darstellung (das „pseudo-polare“-Koordinatensystem).

Übergabeparameter

Name	Typ	Inhalt
grauBild	[MxN]-Matrix	Graustufenbild.
pupillenMittelpunkt	Koordinate	Mittelpunkt der Pupille.
pupillenRadius	Fließkommawert	Radius der Grenze Pupille/Iris.
skleraRadius1	Fließkommawert	Radius der Grenze Iris/Sklera bezüglich des linken Irisrands.
skleraradius2	Fließkommawert	Radius der Grenze Iris/Sklera bezüglich des rechten Irisrands.
xdim	Integer	Dimension des Endergebnisses in X-Richtung.
ydim	Integer	Dimension des Endergebnisses in Y-Richtung.

Rückgabeparameter

Name	Typ	Inhalt
transBild	[MxN]-Matrix	Die transformierenten Irisdaten als Graustufenbild. Die Dimension des Bildes wird von den Übergabeparametern *xdim* und *ydim* bestimmt.

C.3 S-Modul

C.3.1 getSignature.m

Aufgabe

Berechnung der biometrischen Signatur des Auges. Der Funktion werden sämtliche nötigen Parameter übergeben, die für die Durchführung der Waveletttransformation bestimmend sind. Für die eigentlichen Transformationsschritte wird die Routine `mygabortransform.cpp` verwendet (siehe unten). Nach vollzogener Transformation wird mit der Routine `generateIriscode` aus den Waveletkoeffizienten die binäre Signatur gewonnen.

Übergabeparameter

Name	Typ	Inhalt
transformedImage	[MxN]-Matrix	Die transformierenten Irisdaten als Graustufenbild.
startFrequenz	Fließkommawert	Ausgangswert für die mittlere Frequenz der Gaborwavelets.
startHorzBreite	Fließkommawert	Ausgangswert für die horizontale Größe der Gaborwavelets.
startVertBreite	Fließkommawert	Ausgangswert für die vertikale Größe der Gaborwavelets.
analysePunkte	Koordinaten-Vektor	Vektor, der die Koordinaten der Positionen enthält, für die Waveletkoeffizienten berechnet werden.
anzSchritte	Integer	Anzahl der durchzuführenden Berechnungsschritte. Jeder einzelne Berechnungsschritt entspricht einer anzuwendenden Parametrisierung der Gaborwavelets.
schrittGroesse	Fließkommawert	Größe, die bestimmt in welchem Verhältnis die Inkrementierung der Waveletparameter vorgenommen wird. Die Schrittgröße Eins entspricht einer Veränderung der Parameter in Oktavenschritten.

Rückgabeparameter

Name	Typ	Inhalt
irisCode	binärer Vektor	Die berechnete Signatur.
frequenzen	Vektor	Vektor, der die verwendeten mittleren Frequenzen enthält (Teil des Datenrückkanals zur Evaluation der Parametrisierung).
horzBreiten	Vektor	Vektor mit den verwendeten Parametern für die horizontale Größe der Gaborwavelets (Teil des Datenrückkanals zur Evaluation der Parametrisierung).
vertBreiten	Vektor	Vektor mit den verwendeten Parametern für die vertikale Größe der Gaborwavelets (Teil des Datenrückkanals zur Evaluation der Parametrisierung).

C.3.2 mygabortransform.cpp

Aufgabe

Durchführung eines Schritts der Wavelettransformation. Für eine Parametrisierung der Gaborwavelets wird die Berechnung der Waveletkoeffizienten bezüglich der übergebenen Analysepunkte durchgeführt. Es wird, je nach Parameterübergabe, entweder der Real- oder der Imaginäranteil der Koeffizienten berechnet.

Übergabeparameter

Name	Typ	Inhalt
transformedImage	[MxN]-Matrix	Die transformierenten Irisdaten als Graustufenbild.
mittlereFrequenz	Fließkommawert	die mittlere Frequenz der Gaborwavelets.
horzBreite	Fließkommawert	die horizontale Größe der Gaborwavelets.
vertBreite	Fließkommawert	die vertikale Größe der Gaborwavelets.
analysePunkte	Koordinaten-Vektor	Vektor, der die Koordinaten der Positionen enthält, für die Waveletkoeffizienten berechnet werden.
realAnteil	Boolean	Booleanwert, der bestimmt ob der Real- oder der Imaginäranteil der Waveletkoeffizienten berechnet wird.

Rückgabeparameter

Name	Typ	Inhalt
koeffizientenMatrix	[MxN]-Matrix	die berechneten Waveletkoeffizienten.

C.3.3 generateIriscode.m

Aufgabe

Abbildung der komplexen Waveletkoeffizienten auf den binären Iriscode.

Dem Aufruf dieser Routine muss eine Umformung der Koeffizientenmatrix in einen Vektor, der die Koeffizienten enthält, vorgenommen werden. Die Abbildung der Koeffizienten auf den Iriscode kann schrittweise vorgenommen werden (wie in der angefertigten Implementierung) oder sich der vollständigen Wavelettransformation anschließen.

Übergabeparameter

Name	Typ	Inhalt
koeffizienten	Vektor	Vektor, der die errechneten Waveletkoeffizienten enthält.

Rückgabeparameter

Name	Typ	Inhalt
code	Vektor	binärer Vektor, der den Iriscode enthält.

C.4 D-modul

C.4.1 hamming_abstand_mit_bitmasken.m

Aufgabe

Berechnung des Hamming-Abstandes zweier Bitfolgen (als binäre Vektoren repräsentiert) unter Anwendung zweier zugeordneten Bitmasken, die einzelne Bits der Folgen aus der Abstandsberechnung ausnehmen.

Dabei kommt die in Abschnitt 3.1.1 eingeführte Rechenvorschrift zu Einsatz:

$$\mathrm{HD} = \frac{\|A \otimes B \cap maskA \cap maskB\|}{\|maskA \cap maskB\|} \tag{C.1}$$

Übergabeparameter

Name	Typ	Inhalt
vek1	binärer Vektor	erste Bitfolge.
vek2	binärer Vektor	zweite Bitfolge.
mask1	binärer Vektor	erste Bitmaske.
mask2	binärer Vektor	zweite Bimaske.

Rückgabeparameter

Name	Typ	Inhalt
hammingabstand	Fließkommawert	Errechneter Hamming-Abstand.

C.4.2 gesamtauswertung.m

Aufgabe

Für die Iriscodes eines Testdatensatzes von Irisbildern, werden die Hamming-Abstände berechnet. Aus diesen Abständen werden (bezogen auf die Klassifizierung der Bilder) die minmalen, maximalen und mittleren Hamming-Abstände berechnet (siehe Abschnitt 4.6).

Zum Zweck der Evaluation der Parametrisierung findet neben dem paarweisen Vergleich der vollständigen Iriscodes auch ein Vergleich von Teil-Codes statt, aufgeteilt nach den verschiedenen verwendeten Parametrisierungen der Gaborwavelets. Diese Ergebnisse werden nicht zurückgegeben, sondern werden über die Methoden der Datensammlung (siehe Abschnitt 4.2.1) zur späteren Auswertung abgelegt.

Übergabeparameter

Name	Typ	Inhalt
codes	[MxN]-Matrix	Jede Zeile der Matrix entspricht einem binären Iriscode. So enthält eine [5x1024]-Matrix beispielsweise 5 Iriscodes der Größe 1024.
anzahlBilder	Integer	Anzahl der Irisbilder in dem Testdatensatz.
klassen	Vektor	Vektor, der die Zuordnung der Irisbilder zu den Klassen (1 oder 2) enthält.
frequenzen	Vektor	Vektor, der die verwendeten mittleren Frequenzen enthält.
horzBreiten	Vektor	Vektor mit den verwendeten Parametern für die horizontale Größe der Gaborwavelets
vertBreiten	Vektor	Vektor mit den verwendeten Parametern für die vertikale Größe der Gaborwavelets
bitcodeLaengeProWavelet	Integer	Anzahl der Bits des Iriscodes, die aus einer Waveletparametrisierung resultieren.

Rückgabeparameter

Name	Typ	Inhalt
min1	Fließkommawert	Minimaler Hamming-Abstand innerhalb der Klasse 1.
max1	Fließkommawert	Maximaler Hamming-Abstand innerhalb der Klasse 1.
mean1	Fließkommawert	Mittlerer Hamming-Abstand innerhalb der Klasse 1.
min2	Fließkommawert	Minimaler Hamming-Abstand zwischen den Klassen.
max2	Fließkommawert	Maximaler Hamming-Abstand zwischen den Klassen.
mean2	Fließkommawert	Mittlerer Hamming-Abstand zwischen den Klassen.

C.5 Routinen zur Ablaufsteuerung

C.5.1 rahmen.m

Aufgabe

Ablaufsteuerung der Berechnung der Signaturen für einen Testdatensatz verschiedener Irisbilder mit anschließender Auswertung. Der Routine wird, neben den Werten der zu testenden Parametrisierung, ein Pfad zu einer Textdatei übergeben. Diese Textdatei enthält die Pfade der zu testenden Irisbilder und die zugehörige Klassifizierung der Bilder (1 oder 2). Die Ergebnisse werden für eine weitere Auswertung gespeichert.

Das Format einer Zeile der Irisliste ist dabei wir folgt:

`Pfad_des_Irisbilds/Name_des_Irisbildes` (ohne `.bmp`) `[tab]`
`Klasse_des_Irisbildes` (1 oder 2) `[Zeilenende]`

Jede Zeile enthält genau einen Verweis auf ein Irisbild.

Um die Liste einfach zu halten, muss die Datei, mit den zu dem Irisbild gehörenden Metadaten, im selben Verzeichnis liegen und den selben Namen (mit der Erweiterung `.dat`) tragen.

Übergabeparameter

Name	Typ	Inhalt
listname	String	Pfad der Datei, welche die Pfäde zu den Irisbildern enthält.
startFrequenz	Fließkommawert	Ausgangswert für die mittlere Frequenz der Gaborwavelets.
startHorzBreite	Fließkommawert	Ausgangswert für die horizontale Größe der Gaborwavelets.
startVertBreite	Fließkommawert	Ausgangswert für die vertikale Größe der Gaborwavelets.
analysePunkte	Koordinaten-Vektor	Vektor, der die Koordinaten der Positionen enthält, für die Waveletkoeffizienten berechnet werden.
anzSchritte	Integer	Anzahl der durchzuführenden Berechnungsschritte. Jeder einzelne Berechnungsschritt entspricht einer anzuwendenden Parametrisierung der Gaborwavelets.
schrittGroesse	Fließkommawert	Größe, die bestimmt in welchem Verhältnis die Inkrementierung der Waveletparameter vorgenommen wird. Die Schrittgröße Eins entspricht einer Veränderung der Parameter in Oktavenschritten.
show	Boolean	Wenn dieser Parameter auf „true“ gesetzt ist, wird jeder Berechnungsschritt von graphischen Ausgaben begleitet. So kann beispielsweise über ein unmittelbares visuelles Feedback die Qualität der Vorverarbeitung überprüft werden.

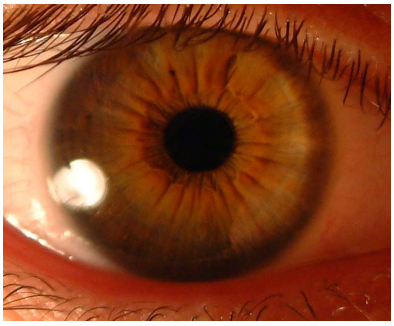
Rückgabeparameter

Name	Typ	Inhalt
codes	[MxN]-Matrix	Jede Zeile der Matrix entspricht einem binären Iriscode. So enthält eine [5x1024]-Matrix beispielsweise 5 Iriscodes der Größe 1024.
anzahlBilder	Integer	Anzahl der Irisbilder in dem Testdatensatz.
klassen	Vektor	Vektor, der die Zuordnung der Irisbilder zu den Klassen (1 oder 2) enthält.
frequenzen	Vektor	Vektor, der die verwendeten mittleren Frequenzen enthält.
horzBreiten	Vektor	Vektor mit den verwendeten Parametern für die horizontale Größe der Gaborwavelets
vertBreiten	Vektor	Vektor mit den verwendeten Parametern für die vertikale Größe der Gaborwavelets
bitcodeLaengeProWavelet	Integer	Anzahl der Bits des Iriscodes, die aus einer Waveletparametrisierung resultieren.

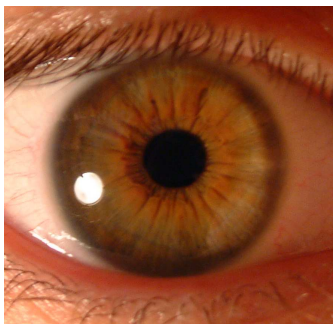
Anhang D

Dokumentation der verwendeten Irisbilder

In den in Kapitel 4 beschriebenen und in Abschnitt E dokumentierten Tests wurde der im folgenden Abgedruckte Satz von Irisbildern verwendet. Die Benennung in der Notation „Bildnummer (Bildklasse)“ entspricht dabei der in Anhang E verwendeten Notation.

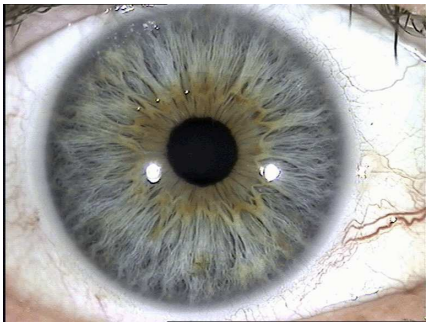


1 (1)

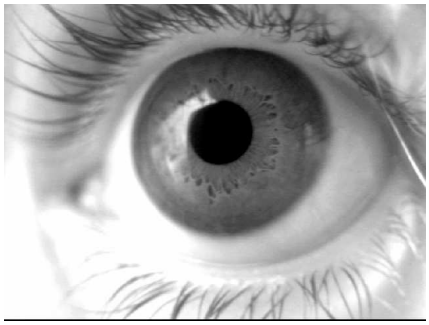


2 (1)

3 (1)

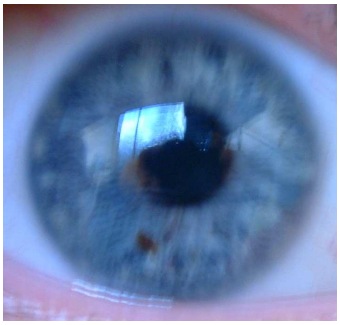


4 (2)

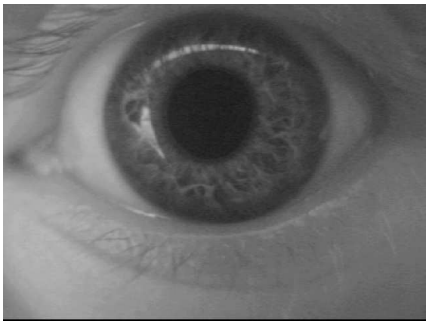


5 (2)

6 (2)



7 (2)



8 (2)

Abbildung D.1: Dokumentation der verwendeten Irisbilder

Anhang E

Details der Testergebnisse

Dieser Teil des Anhangs dokumentiert die Einzelergebnisse der Versuche aus Kapitel 4.

Notation

Die Dokumentation der Einzelergebnisse ist in die Bereiche **Gesamtergebnis** (in den Teilen E.2, E.3 und E.4) und alle **Einzelergebnisse** (in den Teilen E.1, E.2 und E.3) aufgeteilt.

Gesamtergebnis

Die Dokumentation des Gesamtergebnisses eines Versuchs stellt die Zusammenfassung der aufgetretenen Hamming-Abstände aller verwendeten Parametrisierungen dar. Für jede Parametrisierung gibt eine Tabelle den minimalen, maximalen und mittleren Hamming-Abstand, für jeweils die Vergleiche innerhalb der Klasse der Referenziris („Innerhalb Klasse 1“) und zwischen den beiden Klassen („Zwischen Klasse 1/2“), an:

	Min	Max	Mittel
Innerhalb Klasse 1	...	...	...
Zwischen Klasse 1/2	...	...	...

Einzelergebnisse

Die Darstellung eines Einzelergebnisses beginnt mit einer Übersicht der aufgetretenen Hamming-Abstände. Dieses geschieht in einer Tabelle der einzelnen Hamming-Abstände zwischen den Bildern. Die Benennung der Bilder

folgt dabei der Konvention *Bildnummer (Bildklasse)*. Ein Element der Tabelle enthält den jeweiligen Hamming-Abstand zwischen den Bildern, deren Bezeichnungen in der betreffenden Zeile bzw. Spalte stehen:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0	...	...	...	...	...	...	...
2 (1)	...	0	...	...	...	...	...	...
3 (1)	...	...	0	...	...	...	...	...

Die weitere Dokumentation der Einzelergebnisse gliedert sich nach den einzelnen mittleren Frequenzen.

Jeder dieser Abschnitte beginnt mit einer Auflistung der verwendeten Parameter:

ω: ...
α: ...
β: ...

Es folgt eine Tabelle der einzelnen Hamming-Abstände bezüglich des Teils des Iriscodes, der aus den Wavelets mit der angegebenen Parametrisierung generiert wurde. Die Notation entspricht der Tabelle, welche die Hamming-Abstände der vollständigen Iriscodes abbildet:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0	...	...	...	...	...	...	...
2 (1)	...	0	...	...	...	...	...	...
3 (1)	...	...	0	...	...	...	...	...

Der Abschnitt endet mit einer Zusammenfassung, die in einer Tabelle den minimalen, maximalen und mittleren Hamming-Abstand, für jeweils die Vergleiche innerhalb der Klasse der Referenziris („Innerhalb Klasse 1“) und zwischen den beiden Klassen („Zwischen Klasse 1/2“), enthält:

	Min	Max	Mittel
Innerhalb Klasse 1	...	...	...
Zwischen Klasse 1/2	...	...	...

E.1 Grobe Abschätzung

E.1.1 Einzelergebnisse für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 20$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1864	0.2321	0.4079	0.3454	0.3527	0.3583	0.3521
2 (1)	0.1864	0.0000	0.2299	0.4180	0.3477	0.3460	0.3627	0.3454
3 (1)	0.2321	0.2299	0.0000	0.4157	0.3198	0.3382	0.3471	0.3276

Bemerkung E.1 (Gesamtergebnis)
Bei diesem Versuch besitzt das Gesamtergebnis keine Aussagekraft und wurde aus diesem Grund ausgelassen.

Teilergebniss 1

Parameter:

ω: 0.03125
α: 128.00
β: 20.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0938	0.1758	0.4258	0.3242	0.3594	0.4414	0.3086
2 (1)	0.0938	0.0000	0.1680	0.4414	0.3008	0.4062	0.4883	0.3086
3 (1)	0.1758	0.1680	0.0000	0.5000	0.3047	0.4258	0.4062	0.2891

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.145833	0.093750	0.175781
Zwischen Klasse 1/2	0.382031	0.289062	0.500000

Teilergebniss 2

Parameter:

ω: 0.06250
α: 64.00
β: 10.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0977	0.1328	0.4141	0.3789	0.3594	0.3828	0.3750
2 (1)	0.0977	0.0000	0.1445	0.4023	0.3906	0.3945	0.4023	0.3789
3 (1)	0.1328	0.1445	0.0000	0.4375	0.3555	0.3906	0.4453	0.4062

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.125000	0.097656	0.144531
Zwischen Klasse 1/2	0.394271	0.355469	0.445312

Teilergebniss 3

Parameter:

ω: 0.12500
α: 32.00
β: 5.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1211	0.1758	0.4570	0.3672	0.4258	0.4219	0.4180
2 (1)	0.1211	0.0000	0.1875	0.4609	0.3555	0.3984	0.4102	0.3906
3 (1)	0.1758	0.1875	0.0000	0.4453	0.3555	0.4062	0.4180	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.161458	0.121094	0.187500
Zwischen Klasse 1/2	0.407552	0.355469	0.460938

Teilergebniss 4

Parameter:

ω: 0.25000
α: 16.00
β: 2.50

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1875	0.2773	0.4180	0.3906	0.3633	0.3984	0.3711
2 (1)	0.1875	0.0000	0.2773	0.4180	0.4297	0.3945	0.4062	0.4102
3 (1)	0.2773	0.2773	0.0000	0.4297	0.3555	0.3594	0.3320	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.247396	0.187500	0.277344
Zwischen Klasse 1/2	0.390625	0.332031	0.429688

Teilergebniss 5

Parameter:

ω: 0.50000
α: 8.00
β: 1.25

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.2422	0.3242	0.4219	0.3984	0.3867	0.3164	0.3984
2 (1)	0.2422	0.0000	0.3086	0.4375	0.3906	0.3398	0.2930	0.3516
3 (1)	0.3242	0.3086	0.0000	0.4023	0.3477	0.2812	0.2812	0.3086

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.291667	0.242188	0.324219
Zwischen Klasse 1/2	0.357031	0.281250	0.437500

Teilergebniss 6

Parameter:

ω: 1.00000
α: 4.00
β: 0.62

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.2852	0.2656	0.3828	0.2578	0.2734	0.2930	0.3047
2 (1)	0.2852	0.0000	0.2617	0.4102	0.2852	0.2148	0.2578	0.3086
3 (1)	0.2656	0.2617	0.0000	0.3594	0.2656	0.2422	0.2539	0.2188

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.270833	0.261719	0.285156
Zwischen Klasse 1/2	0.288542	0.214844	0.410156

Teilergebniss 7

Parameter:

ω: 2.00000
α: 2.00
β: 0.31

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.2773	0.2734	0.3359	0.3008	0.3008	0.2539	0.2891
2 (1)	0.2773	0.0000	0.2617	0.3555	0.2812	0.2734	0.2812	0.2695
3 (1)	0.2734	0.2617	0.0000	0.3359	0.2539	0.2617	0.2930	0.3047

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.270833	0.261719	0.277344
Zwischen Klasse 1/2	0.292708	0.253906	0.355469

E.2 Feine Abschätzung

E.2.1 Gesamtergebnis

Zusammenfassung für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 20$:

	Mittel	Min	Max
Innerhalb Klasse 1	0.197917	0.140625	0.226562
Zwischen Klasse 1/2	0.427083	0.351562	0.480469

Zusammenfassung für $\omega = 0.041660$, $\alpha = 96$ und $\beta = 15$:

	Mittel	Min	Max
Innerhalb Klasse 1	0.218750	0.171875	0.261719
Zwischen Klasse 1/2	0.394271	0.324219	0.453125

Zusammenfassung für $\omega = 0.062500$, $\alpha = 64$ und $\beta = 10$:

	Mittel	Min	Max
Innerhalb Klasse 1	0.265625	0.203125	0.300781
Zwischen Klasse 1/2	0.375260	0.281250	0.417969

E.2.2 Einzelergebnisse für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 20$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1138	0.1709	0.4326	0.3750	0.3857	0.4268	0.3843
2 (1)	0.1138	0.0000	0.1675	0.4419	0.3657	0.4077	0.4409	0.3818
3 (1)	0.1709	0.1675	0.0000	0.4482	0.3467	0.3955	0.4248	0.3677

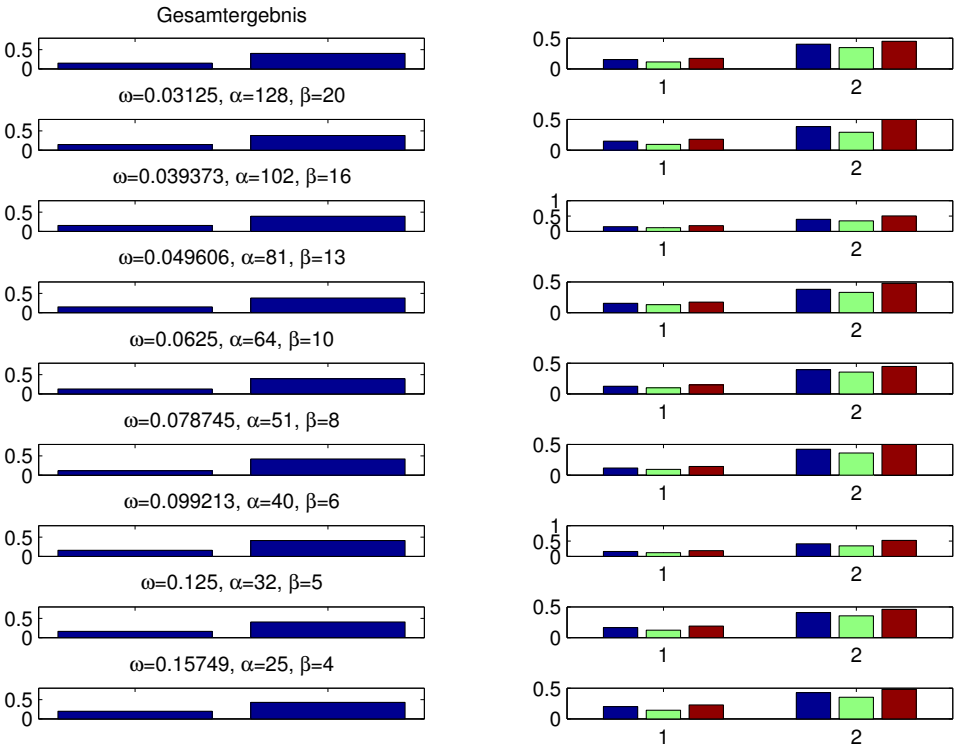


Abbildung E.1: Einzelergebnis 1 der feinen Abschätzung

Teilergebniss 1

Parameter:

ω: 0.03125
α: 128.00
β: 20.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0938	0.1758	0.4258	0.3242	0.3594	0.4414	0.3086
2 (1)	0.0938	0.0000	0.1680	0.4414	0.3008	0.4062	0.4883	0.3086
3 (1)	0.1758	0.1680	0.0000	0.5000	0.3047	0.4258	0.4062	0.2891

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.145833	0.093750	0.175781
Zwischen Klasse 1/2	0.382031	0.289062	0.500000

Teilergebniss 2

Parameter:

ω: 0.03937
α: 101.59
β: 15.87

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1172	0.1875	0.3711	0.3789	0.3516	0.4883	0.3750
2 (1)	0.1172	0.0000	0.1562	0.3945	0.3398	0.3984	0.5039	0.3594
3 (1)	0.1875	0.1562	0.0000	0.4102	0.3477	0.3516	0.4805	0.3438

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.153646	0.117188	0.187500
Zwischen Klasse 1/2	0.392969	0.339844	0.503906

Teilergebniss 3

Parameter:

ω: 0.04961
α: 80.63
β: 12.60

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1289	0.1719	0.3633	0.4219	0.3281	0.4570	0.3750
2 (1)	0.1289	0.0000	0.1523	0.3359	0.3711	0.3320	0.4766	0.3555
3 (1)	0.1719	0.1523	0.0000	0.3711	0.3594	0.3359	0.4727	0.3281

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.151042	0.128906	0.171875
Zwischen Klasse 1/2	0.378906	0.328125	0.476562

Teilergebniss 4

Parameter:

ω: 0.06250
α: 64.00
β: 10.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0977	0.1328	0.4141	0.3789	0.3594	0.3828	0.3750
2 (1)	0.0977	0.0000	0.1445	0.4023	0.3906	0.3945	0.4023	0.3789
3 (1)	0.1328	0.1445	0.0000	0.4375	0.3555	0.3906	0.4453	0.4062

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.125000	0.097656	0.144531
Zwischen Klasse 1/2	0.394271	0.355469	0.445312

Teilergebniss 5

Parameter:

ω: 0.07875
α: 50.80
β: 7.94

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0938	0.1094	0.4766	0.3750	0.4180	0.3672	0.4023
2 (1)	0.0938	0.0000	0.1406	0.5000	0.4141	0.4492	0.3906	0.4180
3 (1)	0.1094	0.1406	0.0000	0.4922	0.3594	0.4492	0.3906	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.114583	0.093750	0.140625
Zwischen Klasse 1/2	0.420312	0.359375	0.500000

Teilergebniss 6

Parameter:

ω: 0.09921
α: 40.32
β: 6.30

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1172	0.1875	0.4961	0.3477	0.4219	0.4023	0.3867
2 (1)	0.1172	0.0000	0.1641	0.5195	0.3711	0.4297	0.3945	0.3945
3 (1)	0.1875	0.1641	0.0000	0.4727	0.3398	0.4141	0.3789	0.3867

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.156250	0.117188	0.187500
Zwischen Klasse 1/2	0.410417	0.339844	0.519531

Teilergebniss 7

Parameter:

ω: 0.12500
α: 32.00
β: 5.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1211	0.1758	0.4570	0.3672	0.4258	0.4219	0.4180
2 (1)	0.1211	0.0000	0.1875	0.4609	0.3555	0.3984	0.4102	0.3906
3 (1)	0.1758	0.1875	0.0000	0.4453	0.3555	0.4062	0.4180	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.161458	0.121094	0.187500
Zwischen Klasse 1/2	0.407552	0.355469	0.460938

Teilergebniss 8

Parameter:

ω: 0.15749
α: 25.40
β: 3.97

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1406	0.2266	0.4570	0.4062	0.4219	0.4531	0.4336
2 (1)	0.1406	0.0000	0.2266	0.4805	0.3828	0.4531	0.4609	0.4492
3 (1)	0.2266	0.2266	0.0000	0.4570	0.3516	0.3906	0.4062	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.197917	0.140625	0.226562
Zwischen Klasse 1/2	0.427083	0.351562	0.480469

E.2.3 Einzelergebnisse für $\omega = 0.041660$, $\alpha = 96$ und $\beta = 15$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1235	0.1846	0.4375	0.3833	0.3936	0.4160	0.3936
2 (1)	0.1235	0.0000	0.1704	0.4468	0.3818	0.4048	0.4233	0.3940
3 (1)	0.1846	0.1704	0.0000	0.4434	0.3491	0.3867	0.4199	0.3867

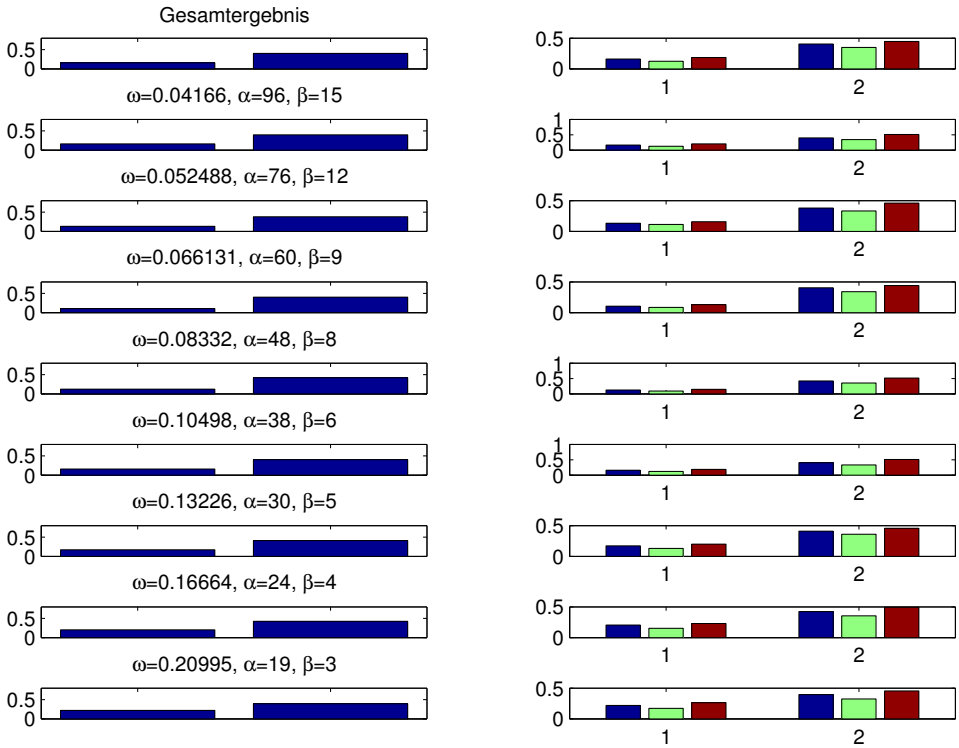


Abbildung E.2: Einzelergebnis 2 der feinen Abschätzung

Teilergebniss 1

Parameter:

ω: 0.04166
α: 96.00
β: 15.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1250	0.2031	0.3672	0.4023	0.3438	0.4922	0.3984
2 (1)	0.1250	0.0000	0.1641	0.3750	0.3789	0.3672	0.5078	0.3594
3 (1)	0.2031	0.1641	0.0000	0.4062	0.3711	0.3438	0.5000	0.3438

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.164062	0.125000	0.203125
Zwischen Klasse 1/2	0.397135	0.343750	0.507812

Teilergebniss 2

Parameter:

ω: 0.05249
α: 76.20
β: 11.91

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1133	0.1562	0.3867	0.4062	0.3359	0.4375	0.3711
2 (1)	0.1133	0.0000	0.1289	0.3594	0.3477	0.3320	0.4570	0.3594
3 (1)	0.1562	0.1289	0.0000	0.3945	0.3359	0.3359	0.4609	0.3555

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.132812	0.113281	0.156250
Zwischen Klasse 1/2	0.378385	0.332031	0.460938

Teilergebniss 3

Parameter:

ω: 0.06613
α: 60.48
β: 9.45

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0859	0.1289	0.4219	0.4062	0.3906	0.3789	0.3789
2 (1)	0.0859	0.0000	0.0977	0.4297	0.3984	0.4141	0.4102	0.3867
3 (1)	0.1289	0.0977	0.0000	0.4414	0.3398	0.4102	0.4219	0.4219

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.104167	0.085938	0.128906
Zwischen Klasse 1/2	0.403385	0.339844	0.441406

Teilergebniss 4

Parameter:

ω: 0.08332
α: 48.00
β: 7.50

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0898	0.1289	0.4844	0.3750	0.4492	0.3594	0.4102
2 (1)	0.0898	0.0000	0.1484	0.5117	0.3867	0.4609	0.3555	0.3984
3 (1)	0.1289	0.1484	0.0000	0.4805	0.3789	0.4766	0.3789	0.4219

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.122396	0.089844	0.148438
Zwischen Klasse 1/2	0.421875	0.355469	0.511719

Teilergebniss 5

Parameter:

ω: 0.10498
α: 38.10
β: 5.95

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1211	0.1680	0.4883	0.3477	0.4062	0.3906	0.4023
2 (1)	0.1211	0.0000	0.1875	0.5078	0.3516	0.4102	0.3945	0.3984
3 (1)	0.1680	0.1875	0.0000	0.4844	0.3281	0.3867	0.3945	0.3750

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.158854	0.121094	0.187500
Zwischen Klasse 1/2	0.404427	0.328125	0.507812

Teilergebniss 6

Parameter:

ω: 0.13226
α: 30.24
β: 4.72

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1289	0.1992	0.4570	0.3789	0.4102	0.4297	0.4102
2 (1)	0.1289	0.0000	0.1875	0.4453	0.3828	0.3906	0.4102	0.4062
3 (1)	0.1992	0.1875	0.0000	0.4453	0.3594	0.3984	0.4336	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.171875	0.128906	0.199219
Zwischen Klasse 1/2	0.409375	0.359375	0.457031

Teilergebniss 7

Parameter:

ω: 0.16664
α: 24.00
β: 3.75

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1523	0.2305	0.4492	0.3750	0.4219	0.4414	0.4141
2 (1)	0.1523	0.0000	0.2266	0.4922	0.3867	0.4570	0.4375	0.4336
3 (1)	0.2305	0.2266	0.0000	0.4766	0.3555	0.4023	0.4062	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.203125	0.152344	0.230469
Zwischen Klasse 1/2	0.423438	0.355469	0.492188

Teilergebniss 8

Parameter:

ω: 0.20995
α: 19.05
β: 2.98

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1719	0.2617	0.4453	0.3750	0.3906	0.3984	0.3633
2 (1)	0.1719	0.0000	0.2227	0.4531	0.4219	0.4062	0.4141	0.4102
3 (1)	0.2617	0.2227	0.0000	0.4180	0.3242	0.3398	0.3633	0.3906

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.218750	0.171875	0.261719
Zwischen Klasse 1/2	0.394271	0.324219	0.453125

E.2.4 Einzelergebnisse für $\omega = 0.062500$, $\alpha = 64$ und $\beta = 10$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1421	0.2061	0.4463	0.3813	0.3950	0.3970	0.3931
2 (1)	0.1421	0.0000	0.2065	0.4536	0.3965	0.4150	0.4062	0.4082
3 (1)	0.2061	0.2065	0.0000	0.4473	0.3501	0.3833	0.3755	0.3931

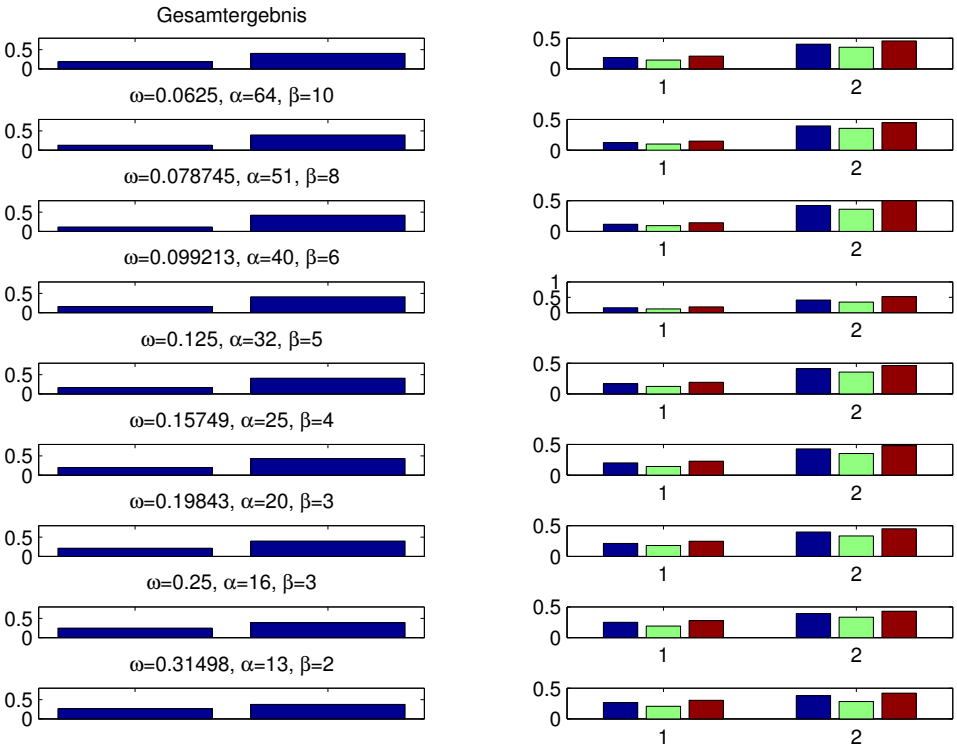


Abbildung E.3: Einzelergebnis 3 der feinen Abschätzung

Teilergebniss 1

Parameter:

ω: 0.06250
α: 64.00
β: 10.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0977	0.1328	0.4141	0.3789	0.3594	0.3828	0.3750
2 (1)	0.0977	0.0000	0.1445	0.4023	0.3906	0.3945	0.4023	0.3789
3 (1)	0.1328	0.1445	0.0000	0.4375	0.3555	0.3906	0.4453	0.4062

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.125000	0.097656	0.144531
Zwischen Klasse 1/2	0.394271	0.355469	0.445312

Teilergebniss 2

Parameter:

ω: 0.07875
α: 50.80
β: 7.94

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0938	0.1094	0.4766	0.3750	0.4180	0.3672	0.4023
2 (1)	0.0938	0.0000	0.1406	0.5000	0.4141	0.4492	0.3906	0.4180
3 (1)	0.1094	0.1406	0.0000	0.4922	0.3594	0.4492	0.3906	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.114583	0.093750	0.140625
Zwischen Klasse 1/2	0.420312	0.359375	0.500000

Teilergebniss 3

Parameter:

ω: 0.09921
α: 40.32
β: 6.30

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1172	0.1875	0.4961	0.3477	0.4219	0.4023	0.3867
2 (1)	0.1172	0.0000	0.1641	0.5195	0.3711	0.4297	0.3945	0.3945
3 (1)	0.1875	0.1641	0.0000	0.4727	0.3398	0.4141	0.3789	0.3867

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.156250	0.117188	0.187500
Zwischen Klasse 1/2	0.410417	0.339844	0.519531

Teilergebniss 4

Parameter:

ω: 0.12500
α: 32.00
β: 5.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1211	0.1758	0.4570	0.3672	0.4258	0.4219	0.4180
2 (1)	0.1211	0.0000	0.1875	0.4609	0.3555	0.3984	0.4102	0.3906
3 (1)	0.1758	0.1875	0.0000	0.4453	0.3555	0.4062	0.4180	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.161458	0.121094	0.187500
Zwischen Klasse 1/2	0.407552	0.355469	0.460938

Teilergebniss 5

Parameter:

ω: 0.15749
α: 25.40
β: 3.97

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1406	0.2266	0.4570	0.4062	0.4219	0.4531	0.4336
2 (1)	0.1406	0.0000	0.2266	0.4805	0.3828	0.4531	0.4609	0.4492
3 (1)	0.2266	0.2266	0.0000	0.4570	0.3516	0.3906	0.4062	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.197917	0.140625	0.226562
Zwischen Klasse 1/2	0.427083	0.351562	0.480469

Teilergebniss 6

Parameter:

ω: 0.19843
α: 20.16
β: 3.15

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1758	0.2461	0.4453	0.3750	0.3984	0.3945	0.3594
2 (1)	0.1758	0.0000	0.2109	0.4492	0.4102	0.4180	0.4062	0.4258
3 (1)	0.2461	0.2109	0.0000	0.4258	0.3320	0.3555	0.3516	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.210938	0.175781	0.246094
Zwischen Klasse 1/2	0.396615	0.332031	0.449219

Teilergebniss 7

Parameter:

ω: 0.25000
α: 16.00
β: 2.50

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1875	0.2773	0.4180	0.3906	0.3633	0.3984	0.3711
2 (1)	0.1875	0.0000	0.2773	0.4180	0.4297	0.3945	0.4062	0.4102
3 (1)	0.2773	0.2773	0.0000	0.4297	0.3555	0.3594	0.3320	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.247396	0.187500	0.277344
Zwischen Klasse 1/2	0.390625	0.332031	0.429688

Teilergebniss 8

Parameter:

ω: 0.31498
α: 12.70
β: 1.98

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.2031	0.2930	0.4062	0.4102	0.3516	0.3555	0.3984
2 (1)	0.2031	0.0000	0.3008	0.3984	0.4180	0.3828	0.3789	0.3984
3 (1)	0.2930	0.3008	0.0000	0.4180	0.3516	0.3008	0.2812	0.3789

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.265625	0.203125	0.300781
Zwischen Klasse 1/2	0.375260	0.281250	0.417969

E.3 Untersuchung des Verhältnisses ω zu α

E.3.1 Gesamtergebnis

Zusammenfassung für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 20$:

	Mittel	Min	Max
Innerhalb Klasse 1	0.197917	0.140625	0.226562
Zwischen Klasse 1/2	0.427083	0.351562	0.480469

Zusammenfassung für $\omega = 0.046875$, $\alpha = 128$ und $\beta = 20$:

	Mittel	Min	Max
Innerhalb Klasse 1	0.317708	0.238281	0.410156
Zwischen Klasse 1/2	0.502344	0.441406	0.574219

E.3.2 Einzelergebnisse für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 20$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1138	0.1709	0.4326	0.3750	0.3857	0.4268	0.3843
2 (1)	0.1138	0.0000	0.1675	0.4419	0.3657	0.4077	0.4409	0.3818
3 (1)	0.1709	0.1675	0.0000	0.4482	0.3467	0.3955	0.4248	0.3677

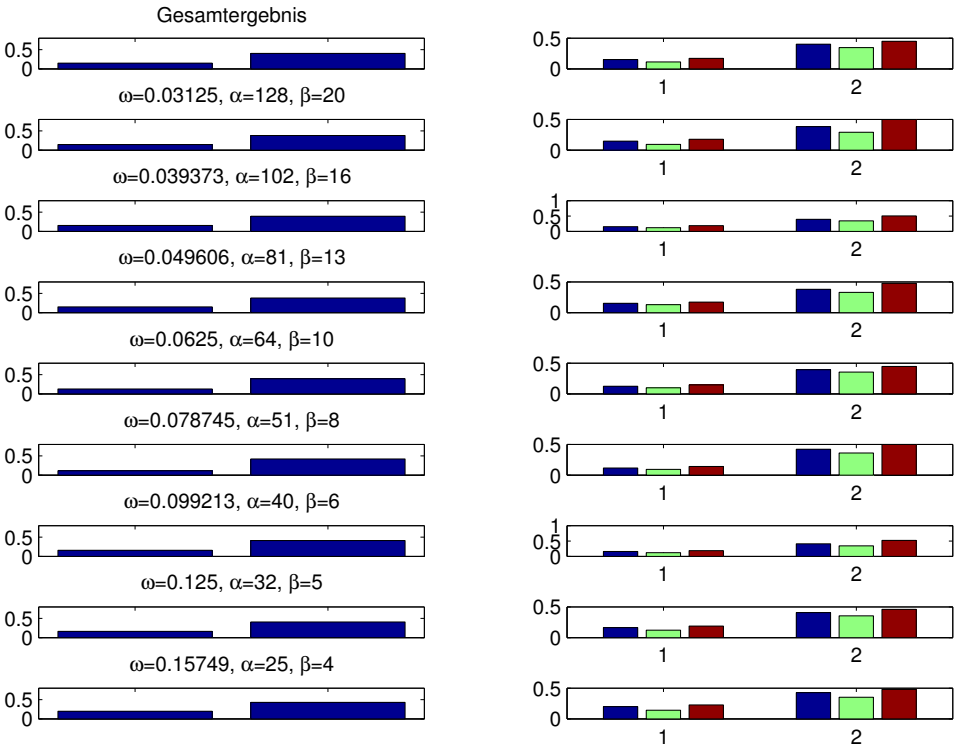


Abbildung E.4: Einzelergebnis 1 der Untersuchung α/ω

Teilergebniss 1

Parameter:

ω: 0.03125
α: 128.00
β: 20.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0938	0.1758	0.4258	0.3242	0.3594	0.4414	0.3086
2 (1)	0.0938	0.0000	0.1680	0.4414	0.3008	0.4062	0.4883	0.3086
3 (1)	0.1758	0.1680	0.0000	0.5000	0.3047	0.4258	0.4062	0.2891

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.145833	0.093750	0.175781
Zwischen Klasse 1/2	0.382031	0.289062	0.500000

Teilergebniss 2

Parameter:

ω: 0.03937
α: 101.59
β: 15.87

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1172	0.1875	0.3711	0.3789	0.3516	0.4883	0.3750
2 (1)	0.1172	0.0000	0.1562	0.3945	0.3398	0.3984	0.5039	0.3594
3 (1)	0.1875	0.1562	0.0000	0.4102	0.3477	0.3516	0.4805	0.3438

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.153646	0.117188	0.187500
Zwischen Klasse 1/2	0.392969	0.339844	0.503906

Teilergebniss 3

Parameter:

ω: 0.04961
α: 80.63
β: 12.60

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1289	0.1719	0.3633	0.4219	0.3281	0.4570	0.3750
2 (1)	0.1289	0.0000	0.1523	0.3359	0.3711	0.3320	0.4766	0.3555
3 (1)	0.1719	0.1523	0.0000	0.3711	0.3594	0.3359	0.4727	0.3281

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.151042	0.128906	0.171875
Zwischen Klasse 1/2	0.378906	0.328125	0.476562

Teilergebniss 4

Parameter:

ω: 0.06250
α: 64.00
β: 10.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0977	0.1328	0.4141	0.3789	0.3594	0.3828	0.3750
2 (1)	0.0977	0.0000	0.1445	0.4023	0.3906	0.3945	0.4023	0.3789
3 (1)	0.1328	0.1445	0.0000	0.4375	0.3555	0.3906	0.4453	0.4062

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.125000	0.097656	0.144531
Zwischen Klasse 1/2	0.394271	0.355469	0.445312

Teilergebniss 5

Parameter:

ω: 0.07875
α: 50.80
β: 7.94

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0938	0.1094	0.4766	0.3750	0.4180	0.3672	0.4023
2 (1)	0.0938	0.0000	0.1406	0.5000	0.4141	0.4492	0.3906	0.4180
3 (1)	0.1094	0.1406	0.0000	0.4922	0.3594	0.4492	0.3906	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.114583	0.093750	0.140625
Zwischen Klasse 1/2	0.420312	0.359375	0.500000

Teilergebniss 6

Parameter:

ω: 0.09921
α: 40.32
β: 6.30

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1172	0.1875	0.4961	0.3477	0.4219	0.4023	0.3867
2 (1)	0.1172	0.0000	0.1641	0.5195	0.3711	0.4297	0.3945	0.3945
3 (1)	0.1875	0.1641	0.0000	0.4727	0.3398	0.4141	0.3789	0.3867

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.156250	0.117188	0.187500
Zwischen Klasse 1/2	0.410417	0.339844	0.519531

Teilergebniss 7

Parameter:

ω: 0.12500
α: 32.00
β: 5.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1211	0.1758	0.4570	0.3672	0.4258	0.4219	0.4180
2 (1)	0.1211	0.0000	0.1875	0.4609	0.3555	0.3984	0.4102	0.3906
3 (1)	0.1758	0.1875	0.0000	0.4453	0.3555	0.4062	0.4180	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.161458	0.121094	0.187500
Zwischen Klasse 1/2	0.407552	0.355469	0.460938

Teilergebniss 8

Parameter:

ω: 0.15749
α: 25.40
β: 3.97

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1406	0.2266	0.4570	0.4062	0.4219	0.4531	0.4336
2 (1)	0.1406	0.0000	0.2266	0.4805	0.3828	0.4531	0.4609	0.4492
3 (1)	0.2266	0.2266	0.0000	0.4570	0.3516	0.3906	0.4062	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.197917	0.140625	0.226562
Zwischen Klasse 1/2	0.427083	0.351562	0.480469

E.3.3 Einzelergebnisse für $\omega = 0.046875$, $\alpha = 128$ und $\beta = 20$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1543	0.2510	0.4990	0.5107	0.4971	0.4922	0.4722
2 (1)	0.1543	0.0000	0.2109	0.4863	0.5137	0.4941	0.4736	0.4780
3 (1)	0.2510	0.2109	0.0000	0.5039	0.4834	0.5029	0.5156	0.4790

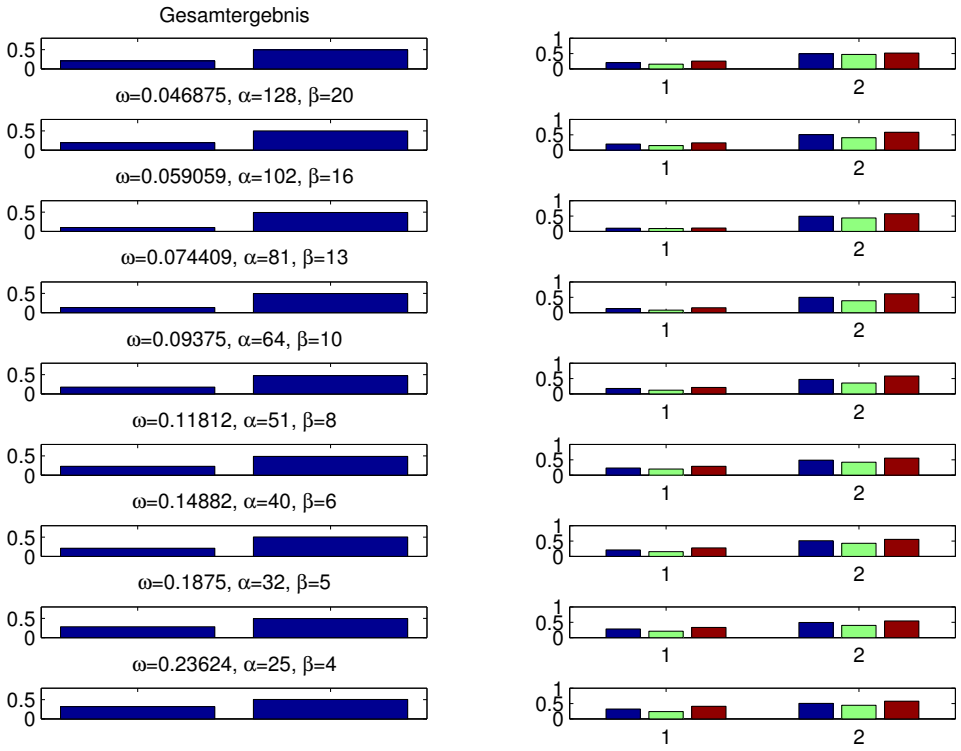


Abbildung E.5: Einzelergebnis 2 der Untersuchung α/ω

Teilergebniss 1

Parameter:

ω: 0.04688
α: 128.00
β: 20.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1445	0.2383	0.5156	0.5781	0.4141	0.5391	0.5352
2 (1)	0.1445	0.0000	0.2031	0.4570	0.5742	0.4023	0.5508	0.4922
3 (1)	0.2383	0.2031	0.0000	0.4492	0.5273	0.4414	0.5273	0.5000

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.195312	0.144531	0.238281
Zwischen Klasse 1/2	0.500260	0.402344	0.578125

Teilergebniss 2

Parameter:

ω: 0.05906
α: 101.59
β: 15.87

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0898	0.1094	0.4844	0.4688	0.4609	0.5195	0.5391
2 (1)	0.0898	0.0000	0.1055	0.4336	0.4648	0.4570	0.4922	0.5430
3 (1)	0.1094	0.1055	0.0000	0.4531	0.4922	0.4766	0.5742	0.5391

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.101562	0.089844	0.109375
Zwischen Klasse 1/2	0.493229	0.433594	0.574219

Teilergebniss 3

Parameter:

ω: 0.07441
α: 80.63
β: 12.60

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0781	0.1523	0.4844	0.5430	0.5625	0.3867	0.4609
2 (1)	0.0781	0.0000	0.1602	0.4844	0.5352	0.5547	0.3945	0.4766
3 (1)	0.1523	0.1602	0.0000	0.5195	0.4453	0.6133	0.4453	0.5117

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.130208	0.078125	0.160156
Zwischen Klasse 1/2	0.494531	0.386719	0.613281

Teilergebniss 4

Parameter:

ω: 0.09375
α: 64.00
β: 10.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1211	0.2031	0.5820	0.4453	0.4883	0.3906	0.4336
2 (1)	0.1211	0.0000	0.2070	0.5781	0.4648	0.5156	0.3555	0.4297
3 (1)	0.2031	0.2070	0.0000	0.5820	0.4141	0.5352	0.4219	0.4414

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.177083	0.121094	0.207031
Zwischen Klasse 1/2	0.471875	0.355469	0.582031

Teilergebniss 5

Parameter:

ω: 0.11812
α: 50.80
β: 7.94

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1992	0.2852	0.5312	0.4570	0.5312	0.5078	0.4297
2 (1)	0.1992	0.0000	0.2031	0.5273	0.4766	0.4727	0.4727	0.4180
3 (1)	0.2852	0.2031	0.0000	0.5352	0.4609	0.4805	0.5508	0.4414

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.229167	0.199219	0.285156
Zwischen Klasse 1/2	0.486198	0.417969	0.550781

Teilergebniss 6

Parameter:

ω: 0.14882
α: 40.32
β: 6.30

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1523	0.2773	0.4492	0.4883	0.5156	0.5234	0.5156
2 (1)	0.1523	0.0000	0.2031	0.4688	0.4922	0.5352	0.5352	0.5117
3 (1)	0.2773	0.2031	0.0000	0.5391	0.5078	0.5273	0.5508	0.4258

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.210938	0.152344	0.277344
Zwischen Klasse 1/2	0.505729	0.425781	0.550781

Teilergebniss 7

Parameter:

ω: 0.18750
α: 32.00
β: 5.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.2109	0.3320	0.4609	0.5312	0.4766	0.5234	0.3984
2 (1)	0.2109	0.0000	0.3008	0.5000	0.5312	0.5312	0.4766	0.4766
3 (1)	0.3320	0.3008	0.0000	0.4727	0.5039	0.4961	0.5430	0.4805

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.281250	0.210938	0.332031
Zwischen Klasse 1/2	0.493490	0.398438	0.542969

Teilergebniss 8

Parameter:

ω: 0.23624
α: 25.40
β: 3.97

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.2383	0.4102	0.4844	0.5742	0.5273	0.5469	0.4648
2 (1)	0.2383	0.0000	0.3047	0.4414	0.5703	0.4844	0.5117	0.4766
3 (1)	0.4102	0.3047	0.0000	0.4805	0.5156	0.4531	0.5117	0.4922

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.317708	0.238281	0.410156
Zwischen Klasse 1/2	0.502344	0.441406	0.574219

E.4 Untersuchung des Verhältnisse α zu β

E.4.1 Gesamtergebnis

Zusammenfassung für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 30$:

	Mittel	Min	Max
Innerhalb Klasse 1	0.200521	0.156250	0.226562
Zwischen Klasse 1/2	0.421354	0.351562	0.476562

Zusammenfassung für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 20$:

	Mittel	Min	Max
Innerhalb Klasse 1	0.197917	0.140625	0.226562
Zwischen Klasse 1/2	0.427083	0.351562	0.480469

Zusammenfassung für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 10$:

	Mittel	Min	Max
Innerhalb Klasse 1	0.200521	0.144531	0.230469
Zwischen Klasse 1/2	0.433594	0.363281	0.496094

E.4.2 Einzelergebnisse für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 30$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1040	0.1665	0.4395	0.3789	0.3857	0.4321	0.3745
2 (1)	0.1040	0.0000	0.1602	0.4351	0.3706	0.4019	0.4443	0.3594
3 (1)	0.1665	0.1602	0.0000	0.4517	0.3501	0.3892	0.4268	0.3594

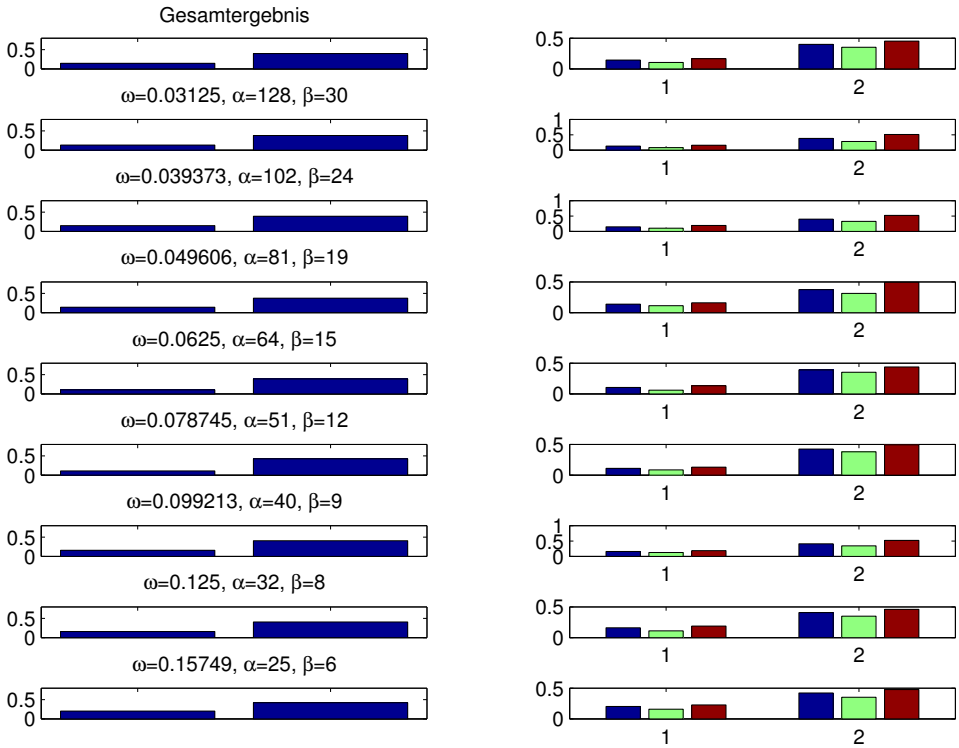


Abbildung E.6: Einzelergebnis 1 der Untersuchung α/β

Teilergebniss 1

Parameter:

ω: 0.03125
α: 128.00
β: 30.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0781	0.1602	0.4258	0.3125	0.3594	0.4688	0.3047
2 (1)	0.0781	0.0000	0.1602	0.4023	0.3047	0.3906	0.4922	0.2812
3 (1)	0.1602	0.1602	0.0000	0.5078	0.2852	0.4258	0.4258	0.2852

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.132812	0.078125	0.160156
Zwischen Klasse 1/2	0.378125	0.281250	0.507812

Teilergebniss 2

Parameter:

ω: 0.03937
α: 101.59
β: 23.81

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1055	0.1914	0.3945	0.3828	0.3555	0.5000	0.3672
2 (1)	0.1055	0.0000	0.1484	0.4062	0.3320	0.3828	0.5195	0.3242
3 (1)	0.1914	0.1484	0.0000	0.4531	0.3477	0.3359	0.4883	0.3320

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.148438	0.105469	0.191406
Zwischen Klasse 1/2	0.394792	0.324219	0.519531

Teilergebniss 3

Parameter:

ω: 0.04961
α: 80.63
β: 18.90

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1133	0.1602	0.3594	0.4141	0.3242	0.4570	0.3750
2 (1)	0.1133	0.0000	0.1406	0.3242	0.3711	0.3125	0.4922	0.3320
3 (1)	0.1602	0.1406	0.0000	0.3711	0.3789	0.3125	0.4766	0.3164

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.138021	0.113281	0.160156
Zwischen Klasse 1/2	0.374479	0.312500	0.492188

Teilergebniss 4

Parameter:

ω: 0.06250
α: 64.00
β: 15.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0586	0.1211	0.4258	0.4219	0.3555	0.3711	0.3555
2 (1)	0.0586	0.0000	0.1328	0.3984	0.4180	0.3828	0.3984	0.3516
3 (1)	0.1211	0.1328	0.0000	0.4297	0.3633	0.3672	0.4375	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.104167	0.058594	0.132812
Zwischen Klasse 1/2	0.390625	0.351562	0.437500

Teilergebniss 5

Parameter:

ω: 0.07875
α: 50.80
β: 11.91

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0859	0.1133	0.4844	0.3789	0.4258	0.3906	0.3906
2 (1)	0.0859	0.0000	0.1289	0.4922	0.4258	0.4570	0.3906	0.4062
3 (1)	0.1133	0.1289	0.0000	0.4648	0.3828	0.4609	0.4023	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.109375	0.085938	0.128906
Zwischen Klasse 1/2	0.423698	0.378906	0.492188

Teilergebniss 6

Parameter:

ω: 0.09921
α: 40.32
β: 9.45

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1250	0.1875	0.5039	0.3516	0.4375	0.4023	0.3750
2 (1)	0.1250	0.0000	0.1562	0.5195	0.3750	0.4297	0.3789	0.3594
3 (1)	0.1875	0.1562	0.0000	0.4805	0.3438	0.4297	0.3555	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.156250	0.125000	0.187500
Zwischen Klasse 1/2	0.408333	0.343750	0.519531

Teilergebniss 7

Parameter:

ω: 0.12500
α: 32.00
β: 7.50

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1094	0.1797	0.4531	0.3711	0.4141	0.4258	0.4180
2 (1)	0.1094	0.0000	0.1875	0.4609	0.3555	0.4062	0.4336	0.4023
3 (1)	0.1797	0.1875	0.0000	0.4453	0.3477	0.3828	0.4180	0.3867

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.158854	0.109375	0.187500
Zwischen Klasse 1/2	0.408073	0.347656	0.460938

Teilergebniss 8

Parameter:

ω: 0.15749
α: 25.40
β: 5.95

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1562	0.2188	0.4688	0.3984	0.4141	0.4414	0.4102
2 (1)	0.1562	0.0000	0.2266	0.4766	0.3828	0.4531	0.4492	0.4180
3 (1)	0.2188	0.2266	0.0000	0.4609	0.3516	0.3984	0.4102	0.3867

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.200521	0.156250	0.226562
Zwischen Klasse 1/2	0.421354	0.351562	0.476562

E.4.3 Einzelergebnisse für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 20$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1138	0.1709	0.4326	0.3750	0.3857	0.4268	0.3843
2 (1)	0.1138	0.0000	0.1675	0.4419	0.3657	0.4077	0.4409	0.3818
3 (1)	0.1709	0.1675	0.0000	0.4482	0.3467	0.3955	0.4248	0.3677

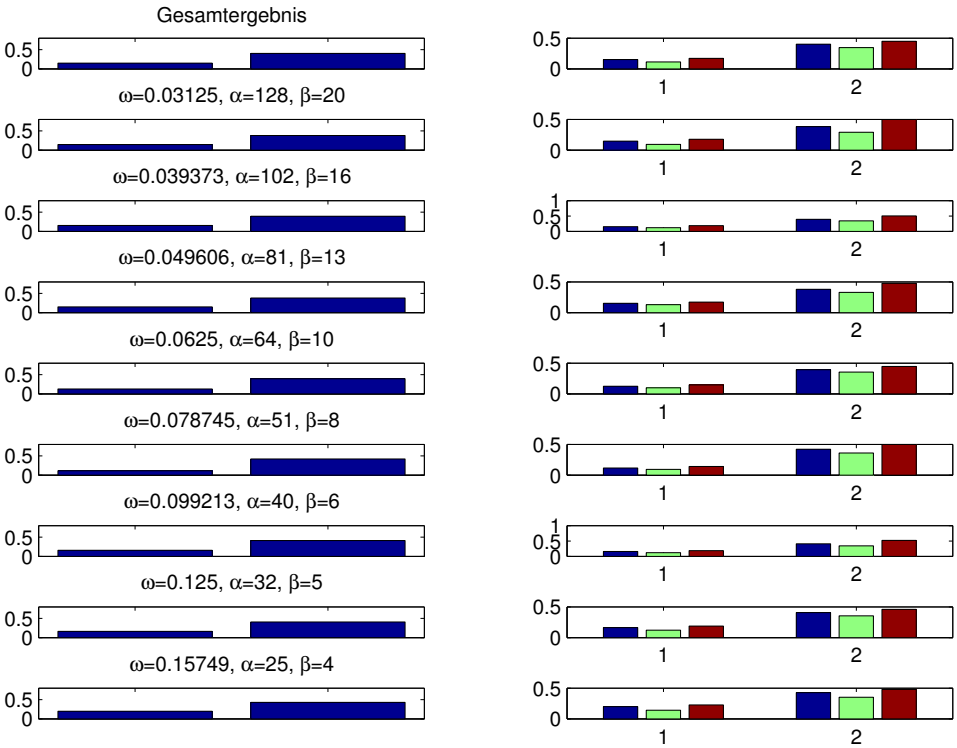


Abbildung E.7: Einzelergebnis 2 der Untersuchung α/β

Teilergebniss 1

Parameter:

ω: 0.03125
α: 128.00
β: 20.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0938	0.1758	0.4258	0.3242	0.3594	0.4414	0.3086
2 (1)	0.0938	0.0000	0.1680	0.4414	0.3008	0.4062	0.4883	0.3086
3 (1)	0.1758	0.1680	0.0000	0.5000	0.3047	0.4258	0.4062	0.2891

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.145833	0.093750	0.175781
Zwischen Klasse 1/2	0.382031	0.289062	0.500000

Teilergebniss 2

Parameter:

ω: 0.03937
α: 101.59
β: 15.87

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1172	0.1875	0.3711	0.3789	0.3516	0.4883	0.3750
2 (1)	0.1172	0.0000	0.1562	0.3945	0.3398	0.3984	0.5039	0.3594
3 (1)	0.1875	0.1562	0.0000	0.4102	0.3477	0.3516	0.4805	0.3438

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.153646	0.117188	0.187500
Zwischen Klasse 1/2	0.392969	0.339844	0.503906

Teilergebniss 3

Parameter:

ω: 0.04961
α: 80.63
β: 12.60

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1289	0.1719	0.3633	0.4219	0.3281	0.4570	0.3750
2 (1)	0.1289	0.0000	0.1523	0.3359	0.3711	0.3320	0.4766	0.3555
3 (1)	0.1719	0.1523	0.0000	0.3711	0.3594	0.3359	0.4727	0.3281

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.151042	0.128906	0.171875
Zwischen Klasse 1/2	0.378906	0.328125	0.476562

Teilergebniss 4

Parameter:

ω: 0.06250
α: 64.00
β: 10.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0977	0.1328	0.4141	0.3789	0.3594	0.3828	0.3750
2 (1)	0.0977	0.0000	0.1445	0.4023	0.3906	0.3945	0.4023	0.3789
3 (1)	0.1328	0.1445	0.0000	0.4375	0.3555	0.3906	0.4453	0.4062

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.125000	0.097656	0.144531
Zwischen Klasse 1/2	0.394271	0.355469	0.445312

Teilergebniss 5

Parameter:

ω: 0.07875
α: 50.80
β: 7.94

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0938	0.1094	0.4766	0.3750	0.4180	0.3672	0.4023
2 (1)	0.0938	0.0000	0.1406	0.5000	0.4141	0.4492	0.3906	0.4180
3 (1)	0.1094	0.1406	0.0000	0.4922	0.3594	0.4492	0.3906	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.114583	0.093750	0.140625
Zwischen Klasse 1/2	0.420312	0.359375	0.500000

Teilergebniss 6

Parameter:

ω: 0.09921
α: 40.32
β: 6.30

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1172	0.1875	0.4961	0.3477	0.4219	0.4023	0.3867
2 (1)	0.1172	0.0000	0.1641	0.5195	0.3711	0.4297	0.3945	0.3945
3 (1)	0.1875	0.1641	0.0000	0.4727	0.3398	0.4141	0.3789	0.3867

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.156250	0.117188	0.187500
Zwischen Klasse 1/2	0.410417	0.339844	0.519531

Teilergebniss 7

Parameter:

ω: 0.12500
α: 32.00
β: 5.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1211	0.1758	0.4570	0.3672	0.4258	0.4219	0.4180
2 (1)	0.1211	0.0000	0.1875	0.4609	0.3555	0.3984	0.4102	0.3906
3 (1)	0.1758	0.1875	0.0000	0.4453	0.3555	0.4062	0.4180	0.3828

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.161458	0.121094	0.187500
Zwischen Klasse 1/2	0.407552	0.355469	0.460938

Teilergebniss 8

Parameter:

ω: 0.15749
α: 25.40
β: 3.97

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1406	0.2266	0.4570	0.4062	0.4219	0.4531	0.4336
2 (1)	0.1406	0.0000	0.2266	0.4805	0.3828	0.4531	0.4609	0.4492
3 (1)	0.2266	0.2266	0.0000	0.4570	0.3516	0.3906	0.4062	0.4023

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.197917	0.140625	0.226562
Zwischen Klasse 1/2	0.427083	0.351562	0.480469

E.4.4 Einzelergebnisse für $\omega = 0.031250$, $\alpha = 128$ und $\beta = 10$

Hamming-Abstände gesamt:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1221	0.1782	0.4360	0.3789	0.3857	0.4194	0.3945
2 (1)	0.1221	0.0000	0.1812	0.4409	0.3613	0.4092	0.4341	0.3965
3 (1)	0.1782	0.1812	0.0000	0.4531	0.3481	0.3940	0.4170	0.3794

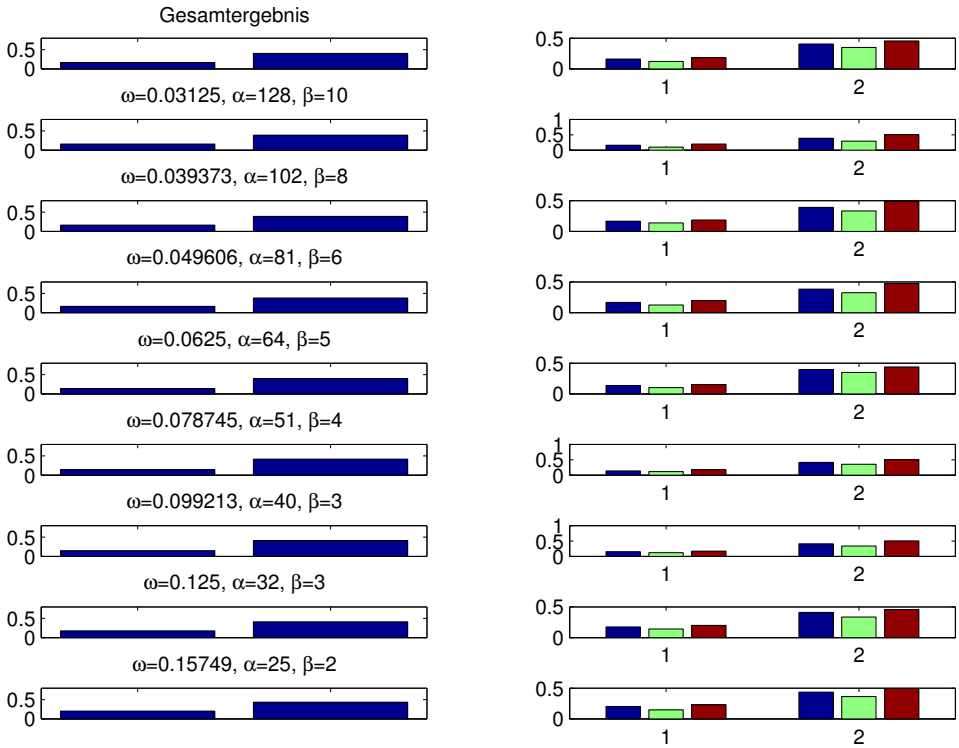


Abbildung E.8: Einzelergebnis 3 der Untersuchung α/β

Teilergebniss 1

Parameter:

ω: 0.03125
α: 128.00
β: 10.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.0977	0.1953	0.4336	0.3398	0.3672	0.4453	0.3398
2 (1)	0.0977	0.0000	0.1758	0.4453	0.2891	0.4023	0.4805	0.3438
3 (1)	0.1953	0.1758	0.0000	0.5039	0.3164	0.4141	0.3906	0.3086

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.156250	0.097656	0.195312
Zwischen Klasse 1/2	0.388021	0.289062	0.503906

Teilergebniss 2

Parameter:

ω: 0.03937
α: 101.59
β: 7.94

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1367	0.1719	0.3789	0.3828	0.3477	0.4531	0.3945
2 (1)	0.1367	0.0000	0.1836	0.3906	0.3320	0.4062	0.4883	0.3672
3 (1)	0.1719	0.1836	0.0000	0.4180	0.3594	0.3555	0.4453	0.3398

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.164062	0.136719	0.183594
Zwischen Klasse 1/2	0.390625	0.332031	0.488281

Teilergebniss 3

Parameter:

ω: 0.04961
α: 80.63
β: 6.30

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1250	0.1953	0.3711	0.4336	0.3242	0.4492	0.3906
2 (1)	0.1250	0.0000	0.1719	0.3477	0.3633	0.3242	0.4727	0.3672
3 (1)	0.1953	0.1719	0.0000	0.3789	0.3477	0.3320	0.4414	0.3672

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.164062	0.125000	0.195312
Zwischen Klasse 1/2	0.380729	0.324219	0.472656

Teilergebniss 4

Parameter:

ω: 0.06250
α: 64.00
β: 5.00

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1016	0.1523	0.4102	0.4023	0.3477	0.3867	0.3828
2 (1)	0.1016	0.0000	0.1523	0.4023	0.3789	0.3867	0.3945	0.3906
3 (1)	0.1523	0.1523	0.0000	0.4375	0.3750	0.3906	0.4375	0.4258

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.135417	0.101562	0.152344
Zwischen Klasse 1/2	0.396615	0.347656	0.437500

Teilergebniss 5

Parameter:

ω: 0.07875
α: 50.80
β: 3.97

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1133	0.1250	0.4766	0.3672	0.4258	0.3516	0.3945
2 (1)	0.1133	0.0000	0.1758	0.5039	0.4023	0.4531	0.3633	0.4297
3 (1)	0.1250	0.1758	0.0000	0.4844	0.3516	0.4414	0.3984	0.3945

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.138021	0.113281	0.175781
Zwischen Klasse 1/2	0.415885	0.351562	0.503906

Teilergebniss 6

Parameter:

ω: 0.09921
α: 40.32
β: 3.15

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1172	0.1719	0.5039	0.3359	0.4297	0.3984	0.3789
2 (1)	0.1172	0.0000	0.1641	0.4961	0.3672	0.4375	0.3906	0.4102
3 (1)	0.1719	0.1641	0.0000	0.4727	0.3359	0.4297	0.3828	0.3789

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.151042	0.117188	0.171875
Zwischen Klasse 1/2	0.409896	0.335938	0.503906

Teilergebniss 7

Parameter:
ω: 0.12500
α: 32.00
β: 2.50

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1406	0.1836	0.4453	0.3789	0.4141	0.4336	0.4258
2 (1)	0.1406	0.0000	0.1992	0.4453	0.3711	0.3984	0.4258	0.4102
3 (1)	0.1836	0.1992	0.0000	0.4570	0.3359	0.3867	0.4297	0.3984

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.174479	0.140625	0.199219
Zwischen Klasse 1/2	0.410417	0.335938	0.457031

Teilergebniss 8

Parameter:
ω: 0.15749
α: 25.40
β: 1.98

Hamming-Abstände:

Bild (Klasse)	1 (1)	2 (1)	3 (1)	4 (2)	5 (2)	6 (2)	7 (2)	8 (2)
1 (1)	0.0000	0.1445	0.2305	0.4688	0.3906	0.4297	0.4375	0.4492
2 (1)	0.1445	0.0000	0.2266	0.4961	0.3867	0.4648	0.4570	0.4531
3 (1)	0.2305	0.2266	0.0000	0.4727	0.3633	0.4023	0.4102	0.4219

Zusammenfassung:

	Mittel	Min	Max
Innerhalb Klasse 1	0.200521	0.144531	0.230469
Zwischen Klasse 1/2	0.433594	0.363281	0.496094

Literaturverzeichnis

[Ashb02] **Julian Ashbourn**: „Biometrics - Advanced Identity Verification“, Springer, London, Berlin, Heidelberg, New York, Barcelona, Hong Kong, Milan, Paris, Singapore, Tokyo, 2002.

[Bäni01] **W. Bäni**: „Wavelets - Eine Einführung für Ingenieure“, Oldenburg Verlag, München, Wien, 2001.

[Bigu01] **Josef Bigun, Fabrizio Smeraldi (Eds.)**: „Audio- and Video-Based Biometric Person Authentication - Third International Conference, Proceedings“, LNCS 2091, Springer, Berlin, Heidelberg, New York, Barcelona, Hong Kong, London, Milan, Paris, Singapore, Tokyo, 2001.

[Bole98] **W. W. Boles, B. Boashah**: „A Human Identification Technique Using Images of the Iris and Wavelet Transform“, IEEE Trans. on Signal Processing, Vol. 46, S. 1185-1188, 1998.

[Boll00] **Ruud M. Bolle, Sharath Pankanti, Nalini K. Ratha**: „Evaluation Techniques for Biometric-Based Authentication Systems (FRR)“, IBM Research Report, IBM T.J. Watson Research Center, RC 21759 (98007), New York, 2000.

[Bovi00] **Al Bovik (Editor)**: „Handbook of Image and Video Processing“, Academic Press, San Diego, San Francisco, New York, Boston, London, Sydney, Tokyo, 2000.

[Broe02] **Arslan Broemme, Marcel Kronberg, Oliver Ellenbeck, Oliver Kasch**: „A Conceptual

Framework for Testing Biometric Algorithms within Operating Systems' Authentication“, SAC 2002, Madrid, 2002.

[Broe02b] **Arslan Broemme**: „A Conceptual Framework for Testing Biometric Algorithms within Operating Systems' Authentication“, Vortragsfolien, ACM Symposium on Applied Computing SAC 2002, Madrid, 2002.

[Cool97] **Ronald Cools, Dirk Laurie, Luc Pluym**: „Algorithm 764: Cubpack++ - A C++ Package for Automatic Two-Dimensional Cubature“, ACM Transactions on Mathematical Software, Vol. 23, No. 1, S. 1 - 15, 1997.

[Corm00] **Lawrence K. Cormack**, „Computational Models of Early Human Vision“, in: [Bovi00], S. 279ff, 2000.

[Daub92] **Ingrid Daubechies**: „Ten Lectures On Wavelets“, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 1992.

[Daug85] **John Daugman**: „Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters“, Journal of the Optical Society of America, Vol. 2, No. 7, July 1985, S. 1160 - 1169, 1985.

[Daug88] **John Daugman**: „Complete discrete 2-d gabor transforms by neural networks for image analysis and compression“, IEEE Transactions On Acoustics, Speech And Signal Processing, 36 (1988),7, S. 1169-1179, 1988.

[Daug93] **John Daugman**: „High Confidence Visual Recognition of Persons by a Test of Statistical Independence“, IEEE Transactions on Pattern Analysis And Machine Intelligence, Vol. 15, No. 11, November 1993, S. 1148 - 1161, 1993.

[Daug99] **John Daugman**: „Recognizing Persons by Their Iris Patterns“, in: [Jain99a], S. 103 - 121, 1999.

[Daug01] **John Daugman, Cathryn Downing**: „Epigenetic Randomness, Complexity and Singularity of Human Iris Patterns“, Proc. R. Soc. London., S. 1737 - 1740, 2001.

[Erol99] **A. Erol, U. Halici, G. Ongun**: „Feature Selective Filtering for Ridge Extraction“, in: [Jain99b], S. 195 - 206, 1999.

[Fisc79] **Gerd Fischer**: „Lineare Algebra“, 5. Auflage, Vieweg Verlag, Braunschweig, 1979.

[Froe03] **Willem Froehling**: „Konzept und exemplarische Implementierung eines gesicherten Kanals zur Übertragung biometrischer Daten“, Studienarbeit, Universität Hamburg, Fachbereich Informatik, Hamburg, 2003.

[Fröh00] **Thorsten Fröhlinghaus**: „Multiscale Gabor Wavelet-based Stereo Disparity Estimation“, Dissertation zur Erlangung des Doktorgrades, Mathematisch-Naturwissenschaftliche Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn, 2000.

[Garc00] **C. Garcia, G. Zikos, G. Tziritas**: „Wavelet Packet Analysis for Face Recognition“, Image and Vision Computing, Vol. 18(4), S. 289 - 297, 2000.

[Giro97] **Bernd Girod, Rudolf Rabenstein, Alexander Stenger**: „Einführung in die Systemtheorie“, B.G. Teubner Verlag, Stuttgart, 1997.

[Goll99] **D. Gollmann**: „Computer Security“, John Wiley & Sons, Chichester, 1999.

[Hama99] **Y. Hamamoto**: „A Gabor Filter-Based Method for Fingerprint Identification“, in: Jain1999, S. 137ff, 1999.

[Hubb97] **Barbara Burke Hubbard**: „Wavelets - Die Mathematik der kleinen Wellen“, Birkhäuser Verlag, Basel, Boston, Berlin, 1997.

[Jain99a] **A.K. Jain, R. Bolle, S.E. Pankanti**: „Biometrics - Personal Identification in Networked Society“, Kluwer Academic Publishers, 1999.

[Jain99b] **L.C. Jain, U. Halici, I. Hayashi, S.B. Lee, S. Tsutsui**: „Intelligent Biometric Techniques in Fingerprint and Face Recognition“, CRC Press, Boca Raton, London, New York, Washington D.C., 1999.

[Kien98] **Uwe Kiencke**: „Signale und Systeme“, R. Oldenburg Verlag, München, Wien, 1998.

[Köni01] **Konrad Königsberger**:„Analysis 1“, Springer, Berlin, Heidelberg, New York, Barcelona, Hong Kong, London, Mailand, Paris, Singapur, Tokio, 2001.

[Krüg01] **Volker Krüger**: „Gabor Wavelet Networks for Object Representation“, Christian-Albrecht-Universität Kiel, Bericht 2014, Kiel, 2001.

[Kuip68] **L. Kuipers, R. Timman**: „Handbuch der Mathematik“, Walter de Gruyter & Co., Berlin, 1968.

[Lim01] **Shinyoung Lim, Kwanyong Lee, Okhwan Byeon, Taiyun Kim**: „Efficient Iris Recognition through Imporovement of Feature Vector and Classifier“, in ETRI Journal, Volume 23, Number 2, June 2001.

[Lips74] **Seymour Lipschutz**: „Probability“, McGraw-Hill International, New York, London, Düsseldorf, Johannesburg, 1974.

[Liu01] **Chengjun Liu, Harry Wechsler**: „Face Recognition Using Independent Gabor Wavelet Features“, in: [Bigu01], S. 20 - 25, 2001.

[Mall98] **Stephane Mallat**: „A Wavelet Tour Of Signal Processing“, Academic Press, San Diego, London, Boston, New York, Sydney, Tokyo, Toronto, 1998.

[Math] **Mathworks**: „Getting started with MATLAB, Version 6“, Rev. for MATLAB 6 (release 12), Mathworks, 2000.

[Meyer98] **Martin Meyer**: „Signalverarbeitung, Analoge und digitale Signale, Systeme und Filter“, Vieweg Verlag, Braunschweig, Wiesbaden, 1998.

[Moul00] **Pierre Moulin**: „Multiscale Image Decomposition and Wavelets“, in: [Bovi00], S. 298ff, 2000.

[Pank00] **Sharanth Pankanti, Anil Jain**: „Fingerprint Classification and Matching“, in: [Bovi00], S. 821 - 835, 2000.

[Patent] **United States Patent**, Biometric Personal Identification System based on Iris Analysis, Inventor: John G. Daugman, Patent Nummber: 5.291.560, Date of Patent: Mar. 1. 1994.

[Pich97] **Olaf Pichler**: „Unüberwachte Bild- und Bildfolgenanalyse mit Mehrkanalfiltermethoden“, Forschungsberichte VDI, Reihe 10: Informatik/Kommunikationstechnik, Nr. 521, 1997.

[Robi02] **Ian Robinson, Michael Hill**: „Algorithm 816: r2d2lri: An Algorithm for Automatic Two-Dimensional Cubature“, ACM Transactions on Mathematical Software, Vol. 28, No. 1, S. 75 - 100, 2002.

[Schr00] **Frank Schröder, Stefan Wörz**: „Theoretische und experimentelle Untersuchungen zur Anwendung der Wavelettransformation auf die Analyse digitaler Signalstrukturen“, Diplomarbeit, Universität Hamburg, Fachbereich Informatik, Arbeitsbereich Kognitive Systeme, Hamburg, 2000.

[Siem91] **Carsten Siemens**: „Theoretische Grundlagen der Bildverarbeitung mit Gabor-Funktionen in biologischen und maschinellen Sehsystemen“, Diplomarbeit, Universität Hamburg, Fachbereich Informatik, 1991.

[Star98] **J.-L. Stark, F. Murtagh, A. Bijaoui**: „Image Processing and Data Analysis - The Multiscale Approach, Cambridge University Press“, Camebridge, New York, Melbourne, 1998.

[Wilc97] **Elke Wilczok**: „Zur Funktionsanalysis der Wavelet- und der Gabortransformation“, Den Naturwissenschaftlichen Fakultäten der Friedrich-Alexander-Universität Erlangen-Nürnberg, 1997.

[Wilc00] **Elke Wilczok**: „New Uncertainty Principles For The Continuous Gabor Transform and The Continuous Wavelet Transform“, in: Documenta Mathematica 5, S. 201 - 226, 2000.

[Wirt99] **Brigitte Wirtz**: „Biometrische Verfahren - Überblick, Evaluierung und aktuelle Themen“, in: Datenschutz und Datensicherheit, 3(23), S. 129 - 134, 1999.

[Wisk99] **L. Wiskott, J.-M. Fellous, N.Krüger und C. Malsburg**: „Face Recognition by Elastic Bunch Graph Matching“, in: [Jain99b], S. 357 - 373, 1999.

[Zhan00] **David D. Zhang**: „Automated Biometrics Technologies and Systems“, Kluwer Academic Publishers Group, 2000.

[Zhu00a] **Yong Zhu, Tieniu Tan, Yunhong Wang**: „Biometric Personal Identification Based on Iris Patterns“, ICPR2000: the 15th International Conference on Pattern Recognition, Barcelona, S. 801 - 804, 2000.

[Zhu00b] **Yong Zhu, Tieniu Tan, Yunhong Wang**: „Biometric Personal Identification Based on Handwriting“, ICPR2000: the 15th International Conference on Pattern Recognition, Barcelona, S. 805 - 808, 2000.